



TUGAS AKHIR - TE 141599

**RANCANG BANGUN PEMANTAU POSISI ARMADA
TRANSPORTASI UMUM PADA *SMART TRANSPORTATION***

MUHAMMAD HILMY ALBANNA
NRP 2213 100 050

Dosen Pembimbing
Dr. Ir. Achmad Affandi, DEA
Dr. Ir. Endroyono, DEA

DEPARTEMEN TEKNIK ELEKTRO
Institut Teknologi Sepuluh Nopember
Surabaya 2017



FINAL PROJECT - TE 141599

***DESIGN OF PUBLIC TRANSPORTATION FLEET
MONITORING POSITION ON SMART TRANSPORTATION***

MUHAMMAD HILMY ALBANNA
NRP 2213 100 050

Lecture Advisor

Dr. Ir. Achmad Affandi, DEA

Dr. Ir. Endroyono, DEA

ELECTRICAL ENGINEERING DEPARTEMENT
Sepuluh Nopember Institute of Technology
Surabaya 2017

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan tugas akhir saya dengan judul “RANCANG BANGUN PEMANTAU POSISI ARMADA TRANSPORTASI UMUM PADA *SMART TRANSPORTATION*” adalah benar – benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan – bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Juli 2017

Muhammad Hilmy Albanna
NRP. 2213 100 050

RANCANG BANGUN PEMANTAU POSISI ARMADA PADA SMART TRANSPORTATION

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik
Pada**

**Bidang Studi Telekomunikasi Multimedia
Departemen Teknik Elektro
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember**

Menyetujui:

Dosen Pembimbing I,

Dosen Pembimbing II,

Dr. Ir. Achmad Affandi, DEA
NIP. 196510141990021001

Dr. Ir. Endrovono, DEA
NIP. 196504041991021001



RANCANG BANGUN PEMANTAUN POSISI ARMADA PADA SMART TRANSPORTATION

Nama : Muhammad Hilmy Albanna
Pembimbing 1 : Dr. Ir. Achmad Affandi, DEA
Pembimbing 2 : Dr. Ir. Endroyono, DEA

ABSTRAK

Teknologi informasi bisa digunakan sebagai sarana pengelolaan Kota Surabaya guna untuk meningkatkan keefektifan pelayanan publik. Salah satu caranya adalah dengan adanya Angkutan Massal Cepat yang akan diterapkan di Kota Surabaya. Armada yang akan diterapkan ini mempunyai beberapa sensor akan yang terintegrasi dengan sistem pemantau yang berada di ruang kontrol untuk memantau pergerakan dari armada, sisa bahan bakar armada, maupun jumlah penumpang yang ada secara realtime. Di dalam armada tersebut terpasang sebuah alat yang disebut dengan OBU, yaitu terminal elektronik yang dapat memberikan informasi mengenai identitas dan posisi AMC, dan menghubungkan komunikasi antara CC-ROOM dengan pengemudi. Tugas akhir ini bertujuan untuk merancang sebuah sistem yang digunakan pada ruang control untuk memantau posisi armada yang sedang beroperasi di jalanan Surabaya

Perancangan dan implementasi dilakukan dengan membuat aplikasi berbasis web dan aplikasi yang akan dipasang di android guna mengambil data GPS dari android tersebut. Aplikasi pengiriman data dari *android* ke *server* dibuat dengan menggunakan PHPStorm dan Andorid Studio. Jaringan yang digunakan yaitu jaringan seluler.

Pengujian sistem pemantau posisi armada ini dilakukan dengan cara perpindahan android dari satu titik ke titik yang lain, dimana hasil yang didapat adalah error posisi dari masing-masing armada yang beroperasi yang mempunyai rata-rata 14 meter dari posisi sebenarnya.

Kata kunci : *Intelligent Transportation System , Delay, Smart City, Tracking, Monitoring*

Halaman ini sengaja dikosongkan

Design of Public Transportation Fleet Monitoring Position on Smart Transportation

Name : Muhammad Hilmy Albanna
Supervisor : Dr. Ir. Achmad Affandi, DEA
Co-Supervisor : Dr. Ir. Endroyono, DEA

ABSTRACT

Technology of information is usable as a management tool for the City of Surabaya to upgrade the effectiveness of its public services. One of the applications is in the form of Fast Mass Transportation that is going to be operated in the city. This fleet possesses several superiorities if compared to the current existing public transportations because it is integrated with a monitoring system in a control room that monitors the fleet's motion, fuel residue, and number of passengers in a real-time way. There is a device installed within the fleet that is called OBU, an electronic terminal that provides information regarding AMC identity and location and puts CC-ROOM in contact with the drivers. This final assignment aims to design a system that can be utilized in a control room to monitor operating fleets' locations on the streets of Surabaya.

The design and implementation process is executed in phases, which are contriving a web based application and installing the application to the androids for GPS data withdrawal. The application used to send data from androids to the server is created with PHPStorm and Android Studio. The kind of network used for the application is cellular.

The fleet's location monitoring system test is done by removing an android from one spot to another, then comparing it to the ideal data that are found theoretically.

Keywords: *Intelligent Transportation System , Delay, Smart City, Tracking, Monitoring*

Halaman ini sengaja dikosongkan

Kata Pengantar

Dengan mengucapkan puji syukur kepada Allah S.W.T., atas limpahan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan buku tugas akhir ini dengan judul:

RANCANG BANGUN PEMANTAUN POSISI ARMADA PADA SMART TRANSPORTATION

Tugas akhir ini disusun sebagai salah satu persyaratan dalam menyelesaikan studi pada bidang studi Telekomunikasi Multimedia di jurusan Teknik Elektro, Institut Teknologi Sepuluh Nopember Surabaya.

Dalam kesempatan ini, penulis ingin menyampaikan rasa terima kasih kepada pihak-pihak yang telah mendukung penulis selama proses menyelesaikan tugas akhir ini, khususnya kepada:

1. Bapak Dr. Ir. Achmad Affandi, DEA dan Bapak Dr. Ir. Endroyono, DEA selaku Dosen Pembimbing atas segala bimbingan selama mengerjakan Tugas Akhir ini.
2. Kedua orang tua penulis, yang selalu memberikan dukungan finansial maupun moral selama penulis menjalani proses perkuliahan di ITS, sampai akhirnya bisa menyelesaikan tugas akhir ini.
3. Kedua adik penulis, Fahmy dan Ghiffari, yang telah menjadi inspirasi dan semangat penulis dalam menyelesaikan tugas akhir. Semoga kalian bisa mencapai pendidikan akademis yang lebih baik.
4. Bapak dan Ibu dosen jurusan teknik elektro ITS, khususnya bidang studi Telekomunikasi Multimedia, atas segala ilmu yang telah diberikan selama penulis kuliah di ITS.
5. Semua rekan-rekan di lab jaringan telekomunikasi, yang telah bekerja dan belajar bersama selama mengerjakan proyek Tugas Akhir ini.
6. Teman-teman Elektro ITS 2013 e53, yang telah menjadi sahabat, keluarga, dan teman seperjuangan selama penulis berkuliah selama 4 tahun di Surabaya.

Dalam penyusunan laporan tugas akhir ini penulis menyadari banyaknya keterbatasan. Oleh karena itu penulis sangat terbuka terhadap kritik dan saran untuk perbaikan karya tugas akhir ini.

Semoga buku Tugas Akhir ini dapat memberikan informasi dan manfaat bagi pembaca pada umumnya dan mahasiswa Jurusan Teknik Elektro bidang Studi Telekomunikasi Multimedia pada khususnya. Dan lebih

jauh diharapkan mampu memberi kontribusi terhadap perkembangan keilmuan, khususnya di bidang telekomunikasi.

Surabaya, Juli 2017

Penulis

Daftar Isi

ABSTRAK	i
ABSTRACT	iii
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	ix
DAFTAR TABEL	xi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan	2
1.5 Metodologi	2
1.6 Sistematika Penulisan	6
1.7 Relevansi dan Manfaat	7
BAB II TINJAUAN PUSTAKA DAN DASAR TEORI	5
2.1 Smart City	5
2.2 Intelligent Transportation System (ITS)	5
2.2.1 Angkutan Massal Cepat (AMC)	6
2.2.2 Automatic Vehicle Location (AVL)	7
2.3 On-Board Unit (OBU)	8
2.4 GPS	9
2.4.1 Faktor yang mempengaruhi error	9
2.5 Jaringan Seluler	10
2.6 Quality of Service	10
2.6.1 Delay	11
2.7 Android	11
2.7.1 Application Layer	11
2.7.2 Application Framework	11
2.8 Personal Home Page	12
2.9 Apache	12
2.10 PHPMyAdmin.....	12
BAB III PERANCANGAN	15
3.1 Perancangan Tampilan Pemantau Posisi Armada.....	15
3.1.1 Rancangan Tampilan Pembaruan Lokasi Armada.....	16
3.1.2 Rancangan Informasi Identitas Armada.....	16

3.1.3 Perancangan Tampilan List Armada	17
3.2 Perancangan On-Board Unit	18
3.3 Perancangan Jaringan Komunikasi	18
3.3.1 Struktur Jaringan Komunikasi Armada.....	18
3.4 Alur Perancangan	19
3.4.1 Proses Pengiriman Data GPS ke Jaringan	19
3.4.2 Proses Pengiriman Data Jaringan ke server	19
3.5 Perangkat Server yang digunakan	19
3.6 Perancangan Sistem Pemantau Posisi Armada	20
3.6.1 Pembuatan Arsitektur Jaringan Pengiriman Data dari GPS	20
3.6.2 Perancangan Database PHPMyAdmin.....	23
3.7 Skenario Kinerja	24
3.7.1 Pembuatan Grid pada Peta Digital	24
3.7.2 Pencatatan Koordinat Teoritis.....	25
3.7.3 Peninjauan Lokasi	26
3.7.4 Pembuatan Tabel Perbandingan.....	27
BAB IV HASIL DAN ANALISA	29
4.1 Pendahuluan	29
4.2 Realisasi Sistem Pemantau Posisi Armada	29
4.2.1 Pembaruan Lokasi Armada.....	29
4.2.2 Informasi Identitas Armada	30
4.2.3 Tampilan Waktu dan Koordinat Saat Data Diterima .	31
4.2.4 Hasil Pembuatan Titik Acuan	31
4.3 Hasil Tampilan Perancangan On-Board Unit	32
4.4 Evaluasi Error Posisi.....	33
4.5 Pengujian dan Analisis <i>Troughput</i>	45
4.6 Pengujian dan Analisis <i>Packet Loss</i>	47
BAB V KESIMPULAN DAN SARAN	44
5.1 Kesimpulan	44
5.2 Saran	44
DAFTAR PUSTAKA	47
LAMPIRAN	49
RIWAYAT HIDUP	65

Table of Content

ABSTRAK	<i>i</i>
ABSTRACT	<i>iii</i>
FOREWORD	<i>v</i>
TABLE OF CONTENT	<i>vii</i>
LIST OF FIGURE	<i>ix</i>
LIST OF TABEL	<i>xi</i>
CHAPTER INTRODUCTION.....	<i>1</i>
1.8 Background	<i>1</i>
1.9 Problems	<i>2</i>
1.10 Scope of Problems	<i>2</i>
1.11 Objective	<i>2</i>
1.12 Methodology	<i>2</i>
1.13 Structure	<i>6</i>
1.14 Relevance	<i>7</i>
CHAPTER II BASIC THEORY	<i>5</i>
2.11 Smart City	<i>5</i>
2.12 Intelligent Transportation System (ITS).....	<i>5</i>
2.2.1 Angkutan Massal Cepat (AMC)	<i>6</i>
2.2.2 Automatic Vehicle Location (AVL)	<i>7</i>
2.13 On-Board Unit (OBU)	<i>8</i>
2.14 GPS	<i>9</i>
2.4.2 Factors of Error.....	<i>9</i>
2.15 Cellular Network	<i>10</i>
2.16 Quality of Service	<i>10</i>
2.6.1 Delay.....	<i>11</i>
2.17 Android	<i>11</i>
2.7.1 Application Layer.....	<i>11</i>
2.7.2 Application Framework	<i>11</i>
2.18 Personal Home Page	<i>12</i>
2.19 Apache	<i>12</i>
2.20 PHPMyAdmin	<i>12</i>
CHAPTER III DESIGN	<i>15</i>
3.8 Design of Fleet Position Monitoring Interface	<i>15</i>
3.1.1 Design of Fleet Update Location.....	<i>16</i>
3.1.2 Design of Fleet Information.....	<i>16</i>

3.1.3 Design of List's Fleet.....	17
3.9 Design of On-Board Unit	18
3.10 Design of Communication Network.....	18
3.3.1 Fleet's Communication Structure.....	18
3.11 Design of System.....	19
3.4.3 Transmitting Process GPS's Data to Network.....	19
3.4.4 Transmitting Process Data's Network to Server	19
3.12 Server Device Used	19
3.13 Design of System Monitoring Fleet Position.....	20
3.6.1 Architecture of Network Transmitting Data from.....	
GPS	20
3.6.2 Design of Database PHPMyAdmin	23
3.14 Scenario of Performance.....	24
3.7.1 Grid Making on Digital Map.....	24
3.7.2 Reference Coordinate	25
3.7.3 Location Review.....	26
3.7.4 Comparison Table.....	27
CHAPTER RESULT AND ANALISIS	29
4.1 Introduction	29
4.2. Realisation of Fleet Monitoring System.....	29
4.2.1 Location Update.....	29
4.2.2 Fleet Information.....	30
4.2.3 Time and Coordinate Interface.....	31
4.2.4 Result of Reference Point	31
4.3 Result of On-Board Unit Interface	32
4.4 Error Position Evaluation	33
4.5 Troughput Analysis	45
4.6 Packet Loss Analysis	47
CHAPTER V CONCLUSION AND RECOMMENDATION	44
5.3 Conclusion	44
5.4 Recommendation.....	44
REFERENCE	47
APPENDIX	49
BIOGRAPHY	65

DAFTAR GAMBAR

Gambar 2.1 Prinsip Kerja AVL	8
Gambar 2.2 Desain OBU	8
Gambar 3.1 Desain Tampilan Web Pemantau Armada Bus	15
Gambar 3.2 Tampilan Pembaruan Lokasi Armada	16
Gambar 3.3 Tampilan Informasi Armada	17
Gambar 3.4 Rancangan Tampilan List Armada	17
Gambar 3.5 Desain Tampilan APK Android.....	18
Gambar 3.6 Diagram blok pengiriman data GPS	19
Gambar 3.7 Arsitektur jaringan	20
Gambar 3.8 Diagram Protokol Client to Server	15
Gambar 3.9 Diagram Protokol Dari Ruang kontrol ke server	15
Gambar 3.10 Tampilan Struktur dari tabel kendaraan.....	16
Gambar 3.11 Cuplikan Grid Untuk Pengujian.....	17
Gambar 3.12 Koordinat teoritis di depan RSIA Kendangsari.....	18
Gambar 3.13 Peninjauan Lokasi RSIA Kendangsari MERR.....	20
Gambar 4.1 Tampilan awal pemantau posisi armada	20
Gambar 4.2 Tampilan A	21
Gambar 4.3 Tampilan B.....	22
Gambar 4.4 Tampilan identitas armada.....	25
Gambar 4.5 Tampilan waktu penerimaan data	25
Gambar 4.6 Tampilan List Armada yang akan dipantau	26
Gambar 4.7 Titik acuan	27
Gambar 4.8 Tampilan pada Android sebagai fungsi GPS	27
Gambar 4.9 Grafik Error terhadap acuan titik 1	28
Gambar 4.10 Grafik Error terhadap acuan titik 2	29
Gambar 4.11 Grafik Error terhadap acuan titik 3	29
Gambar 4.12 Grafik Error terhadap acuan titik 4	35
Gambar 4.13 Grafik Error terhadap acuan titik 5	30
Gambar 4.14 Grafik Error terhadap acuan titik 6	31
Gambar 4.15 Grafik Error terhadap acuan titik 7	32
Gambar 4.16 Grafik Error terhadap acuan titik 8	33
Gambar 4.17 Tampilan di lokasi pengukuran	35
Gambar 4.18 Tampilan yang terbaca pada display	35
Gambar 4.19 Troughput pada Wifi	44
Gambar 4.20 Trouhput pada Jaringan Seluler	44
Gambar 4.21 Packet Loss pada Wifi	45
Gambar 4.22 Packet Loss pada Jaringan Seluler	45

DAFTAR TABEL

Tabel 3.1 Contoh tabel perbandingan yang akan digunakan	27
Tabel 4.1 Hasil Pengukuran pada lokasi rumah sakit kendangsari ..	48
Tabel 4.2 Hasil Pengukuran pada lokasi rumah sakit kendangsari ..	33
Tabel 4.3 Hasil Pengukuran pada lokasi ATM BNI Unair	34
Tabel 4.4 Hasil Pengukuran pada lokasi Galaxy Mall	35
Tabel 4.5 Hasil Pengukuran pada lokasi Hartono Elektronik	36
Tabel 4.6 Hasil Pengukuran pada lokasi Kampus Stikom.....	37
Tabel 4.7 Hasil Pengukuran pada lokasi Bebek Harissa	38
Tabel 4.8 Hasil Pengukuran pada lokasi BNI Kampus UPN	39
Tabel 4.9 Data Pengukuran Gabungan	39

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Teknologi telekomunikasi dan informasi di Indonesia sedang dalam masa perkembangan yang cukup signifikan terutama semua hal yang menyangkut masalah publik. Kota Surabaya mengalami peningkatan jumlah kendaraan pribadi yang besar dari tahun ke tahun sehingga dapat menyebabkan kepadatan lalu lintas di ruas jalan raya Surabaya. Salah satu faktornya adalah kurangnya moda transportasi publik yang memadai. Badan Perencanaan Pembangunan Nasional telah memperkirakan Angkutan Massal Cepat (AMC) Kota Surabaya beroperasi pada tahun 2018. Proyek AMC Kota Surabaya telah masuk dalam Rancangan Program Jangka Menengah Nasional Bappenas tahun 2015 – 2019. Hal ini menunjukkan bahwa pemerintah kota Surabaya telah berkomitmen untuk membangun sebuah angkutan massal cepat untuk mengurangi kemacetan di Kota Surabaya [1].

Angkutan Massal Cepat memiliki sistem komunikasi terintegrasi dengan wujud pemasangan *On-Board Unit* (OBU) pada setiap armada dan sistem *Passanger Information Board* (PIB) pada halte. Dari armada yang telah direncanakan rutanya tentu memerlukan laporan secara *real time* mengenai keberadaan dari masing-masing armada yang harus terhubung dengan ruang control agar operator pada *control room* dapat selalu memantau keberadaan posisi dan informasi mengenai armada yang sedang beroperasi, maka dalam tugas akhir ini dibuatlah sistem yang dapat menunjukkan keberadaan, rute, serta pemberhentian terdekatnya secara *real time* dan dapat langsung disampaikan kepada penumpang yang berada di dalam armada maupun yang di halte [2]

1.2 Perumusan Masalah

1. Bagaimana cara mengetahui dan menampilkan posisi setiap armada secara realtime?
2. Bagaimana cara memastikan posisi dari armada?

1.3 Batasan Masalah

1. GPS yang digunakan adalah GPS untuk masyarakat sipil
2. Lokasi yang digunakan hanya area jalan MERR
3. Tampilan yang dibangun dari sisi ruang kontrol

1.4 Tujuan Tugas Akhir

1. Terdapat sistem yang menghubungkan antara OBU pada armada dan ruang kontrol, sehingga dapat mengetahui semua informasi mengenai rute perjalanan.
2. Mengimplementasikan dari Google Map API pada sistem yang telah dibuat.

1.5 Metodologi

1. Studi Literatur

Mempelajari buku serta referensi mengenai *Intelligent Transportation System*, *Smart City*, Sistem Tracking, dan *On-Board Unit* (OBU).

2. Perancangan pada software PHPStorm

Membuat sistem pemrograman yang dapat menghubungkan GPS dan CCROOM secara *real time*. Kemudian mengintegrasikan dengan Google Map API agar mendapatkan titik posisi dari armada.

3. Pengkodean

Tahap penerjemahan data atau pemecahan masalah yang telah dirancang ke dalam bahasa pemrograman.

4. Pengujian Data

Melakukan pengujian fungsi dari sistem yang telah dibuat kemudian menguji performansi dari sistem tersebut yang melibatkan jaringan seluler dan GPS.

5. Analisis Data

Dilakukan terhadap hasil dari pengujian sehingga dapat ditentukan tingkat error, delay, dan kelayakan sistem yang telah dibuat sehingga sistem dapat bekerja dengan normal.

6. Kesimpulan

Membuat kesimpulan yang dapat diambil dari hasil analisa dari data yang telah didapatkan dari tugas akhir ini untuk menjawab dari permasalahan yang dihadapi.

7. Penyusunan Laporan

Membuat dokumentasi pelaksanaan tugas akhir yang meliputi dasar teori, proses perancangan, pembuatan, dan pengujian.

1.6 Sistematika Laporan

Penulisan dibagi dalam 5 bab, dengan sistematika sebagai berikut:

BAB 1: PENDAHULUAN

Bab ini menjelaskan tentang latar belakang permasalahan dan Batasan masalah, tujuan, metodologi serta relevansi dari penulisan tugas akhir.

BAB 2: DASAR TEORI

Pada bab ini menjelaskan teori yang berhubungan dengan kegiatan tugas akhir antara lain angkutan massal umum, OBU, android, dan jaringan seluler serta penjelasan mengenai Bahasa pemrograman yang dipakai yaitu PHP.

BAB 3: METODOLOGI

Meliputi pembahasan mengenai metodologi yang dilakukan dalam pengerjaan tugas akhir ini, termasuk menjelaskan *flowchart* yang akan melalui jaringan seluler.

BAB 4: ANALISA DATA DAN PEMBAHASAN

Pada bab ini akan ditampilkan data hasil pengukuran dan analisa dari hasil pengukuran tersebut, mulai dari tingkat kepresisian hingga delay yang ada.

BAB 5: KESIMPULAN DAN SARAN

Menjelaskan mengenai kesimpulan yang didapat dalam penelitian tugas akhir ini serta merekomendasikan saran yang bisa diterapkan agar dapat lebih baik untuk kedepannya dalam pemantauan armada pada *Smart City* ini.

1.7 Relevansi

Hasil penelitian dari tugas akhir ini diharapkan dapat membantu terbentuknya salah satu sistem yang menunjang pemantauan posisi armada secara *real time*. Lebih mengefisiensikan transportasi massal umum yang akan beroperasi di Kota Surabaya.

BAB 2

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1. *Smart City*

Smart City atau secara harfiah yang berarti kota pintar, berakar pada implementasi informasi dan komunikasi yang *user-friendly*. Teknologi yang dikembangkan oleh industri besar untuk ruang perkotaan, merupakan sebuah konsep penerapan teknologi yang digunakan pada suatu kota sebagai sebuah interaksi dari berbagai sistem yang ada di dalam kota tersebut. Konsep *smart city* ini awalnya diterapkan di Amerika Serikat dan Uni Eropa yang pada awalnya konsep ini bertujuan untuk meningkatkan layanan publik. Saat ini sudah banyak negara-negara yang sudah menerapkan konsep *smart city* seperti di benua Amerika, Australia, Eropa, dan bahkan Asia. Di Indonesia sendiri, *smart city* ini pertama kali diterapkan untuk kota Bandung. Jakarta sebagai ibukota juga perlahan sedang melakukan masa transisi untuk menerapkan konsep *smart city* ini mengingat Jakarta merupakan magnet yang sangat kuat untuk menarik turis, investor, perdagangan, dan lainnya. Sebagai kota kedua terbesar di Indonesia, Surabaya juga akan menerapkan konsep *smart city* ini. Salah satu bidang yang mendukung konsep ini yaitu rancangan sistem pemantau posisi armada untuk mendukung *Smart Transportation*. Surabaya berencana untuk membuat Angkutan Massal Cepat (AMC) yang berbasis Intelligent Transportation System (ITS). Pada penelitian ini, direncanakan bahwa AMC akan terhubung selalu dengan internet untuk memberikan informasi *real time* seputar keberadaan AMC tersebut kepada pengguna. Data-data tersebut akan disimpan pada suatu server yang menggunakan konsep Cloud Computing dan nantinya dapat diketahui secara langsung oleh pengguna. Nantinya, setiap pengguna akan terintegrasi dengan aplikasi yang dapat diunduh pada smartphone masing-masing sehingga dapat memudahkan untuk mencari informasi mengenai keberadaan dari AMC tersebut.

2.2. *Intelligent Transportation System (ITS)*

ITS merupakan sistem transportasi yang mengombinasikan antara komunikasi, komputer dan teknologi pengawasan yang telah dikembangkan dan diterapkan pada daerah transportasi untuk meningkatkan sistem performansi, keamanan, efisiensi, produktivitas, tingkat layanan, dampak lingkungan, konsumsi energi, dan mobilitas.

Sistem transportasi cerdas dibagi menjadi tiga bidang :

- Infrastruktur cerdas (*Intelligent Infrastructure*), contohnya sinyal lalu lintas di jalan raya, *variable message system* untuk mengingatkan pengguna jalan dari bahaya yang terjadi di depan, dan sinyal ramp jalan bebas hambatan yang bekerja untuk menjaga kelancaran jalan raya
- Kendaraan cerdas (*Smart Vehicle*), seperti notifikasi tabrakan otomatis, pemberitahuan batas kecepatan, peringatan tabrakan mundur dan depan, dan sistem navigasi GPS.
- Layanan informasi (*Information Services*), contohnya informasi kedatangan bus pada ponsel, sistem navigasi dalam mobil yang menerima kondisi lalu lintas terkini untuk panduan kemacetan, dan program akses nasional untuk truk [2]

Pada tugas akhir ini bagian yang akan dikembangkan adalah mengetahui posisi dari armada bus yang sedang beroperasi, sehingga pemantauan lebih mudah dilakukan pada ruang kontrol yang telah dibangun.

2.2.1 Angkutan Massal Cepat (AMS)

Angkutan massal cepat merupakan suatu sistem transportasi umum transit cepat yang sedang dibangun di Surabaya. Dalam upaya mengatasi permasalahan transportasi di Surabaya diantaranya yaitu kemacetan, keterbatasan lahan untuk pembangunan jalan, perjalanan penumpang yang tidak efisien, dan meningkatnya polusi udara atas penggunaan kendaraan pribadi maka dibutuhkan upaya penyediaan angkutan umum massal yang merupakan sarana transportasi berkelanjutan [3]. Angkutan Massal Cepat ini akan ditanamkan sebuah modul yang dinamakan *On-Board Unit* (OBU), selain itu agar angkutan massal cepat ini dapat terpantau dengan baik maka diterapkanlah *Automatic Vehicle Location* (AVL). Di sisi lain, ketersediaan Angkutan Massal Cepat menghubungkan prasarana transportasi utama di Kota Surabaya diharapkan mampu meningkatkan mobilitas orang secara efisien terhadap wilayah Surabaya dan sekitarnya.

Tujuan dikembangkannya Angkutan Massal Cepat adalah :

- Meningkatkan kualitas layanan dan keselamatan lalu-lintas
- Menurunkan tingkat kemacetan
- Meningkatkan efisiensi transportasi
- Mengurangi polusi udara
- Meningkatkan efisiensi pemanfaatan energi

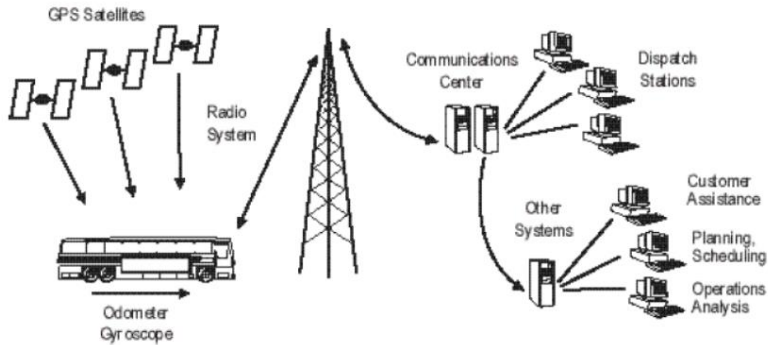
Sistem angkutan massal yang umum dipergunakan di perkotaan adalah sebagai berikut :

- *Bus Rapid Transit* (BRT) – Teknologi berbasis Bus, pada umumnya beroperasi pada jalur khusus yang sebidang dengan permukaan jalan yang ada, pada kondisi tertentu (misalnya, persimpangan atau pusat kota) yang diperlukan pemisahan elevasi, BRT dilewatkan terowongan atau jembatan khusus.
- *Light Rail Transit* (LRT) - Teknologi berbasis Rel-Listrik, pada umumnya beroperasi menggunakan kendaraan rel tunggal atau kereta listrik pendek di jalur rel khusus sebidang dengan permukaan tanah dengan konektor listrik di atas kendaraan. Jenis lain dari LRT adalah *Tram System*, pada umumnya dengan ukuran kendaraan yang lebih kecil dan beroperasi di jalur jalan raya tanpa pemisahan dengan lalu-lintas lainnya.
- *Personal Rapid Transit* (PRT) - Teknologi berbasis rel atau roda, mengangkut penumpang dengan kendaraan berfasilitas AVG (*automatic guided vehicles*) yang beroperasi pada jalur khusus [4].

2.2.2 Automatic Vehicle Location (AVL)

AVL merupakan sistem pelacak kendaraan yang berbasis komputer. Masing masing kendaraan dapat diketahui posisinya kemudian dilanjutkan ke *control room*. Data informasi yang menunjukkan posisi armada yang berlangsung beberapa detik kemudian diteruskan ke control room yang berbentuk data mentah. Teknologi yang telah berkembang saat ini yang mempunyai manfaat melacak kendaraan pada sistem AVL adalah *Global Position System* (GPS). Yang menggunakan sinyal yang ditransmisikan dari

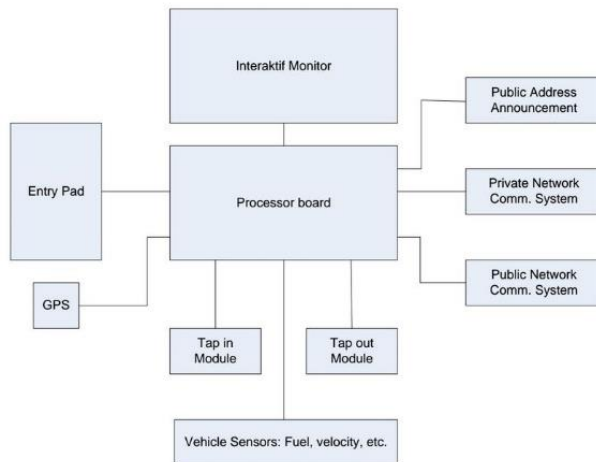
jaringan 24 satelit yang mengorbit.



Gambar 2.1 Prinsip Kerja AVL

2.3 *On-Board Unit (OBU)*

Program aplikasi OBU dikembangkan dengan bahasa Linux C dan Skrip Shell. Multiproses, *shared memory*, dan Socket pemrograman yang biasa digunakan untuk menggambarkan informasi GPS yang lebih jelas. *Interface* yang digunakan adalah perangkat lunak pemrograman Qtopia [1].



Gambar 2.2 Desain OBU

Pada gambar 2.2 dijelaskan beberapa perangkat yang terintegrasi sehingga menjadi OBU, dalam tugas akhir ini akan dibahas mengenai GPS,

karena alat tersebut merupakan peranan utama dalam penentuan lokasi dari masing-masing armada angkutan massal cepat di Surabaya yang nantinya data dari GPS tersebut akan dikirimkan ke server melalui jaringan seluler, untuk mendapatkan fungsi yang sama dengan modul GPS maka digunakanlah GPS pada *mobile phone* android.

2.4 GPS

Global positioning System adalah satu-satunya system navigasi global (GNSS) yang berfungsi penuh. Sistem ini dibuat dan dikembangkan oleh NAVSTAR GPS (Navigation Satellite And Ranging).

Dengan konstelasi sedikitnya 24 buah (tidak termasuk 3 satelit cadangan jika gagal kerja) satelit orbit bumi jarak menengah seperti yang diperlihatkan gambar dibawah. Mentransmisikan sinyal gelombang mikro secara tepat, sistem ini memungkinkan sebuah penerima GPS secara luas dipergunakan sebagai alat bantu untuk navigasi, berguna pula untuk pembuatan peta, survey permukaan bumi , perdagangan dan berbagai penelitian. Kinerja dari GPS yang diharapkan adalah error dari penentuan lokasi hampir mendekati nol sampai dua meter, sedangkan jika diatas 20 meter maka kinerja dari GPS tersebut sudah mengalami error yang tidak bisa ditoleransi.

2.4.1 Faktor yang Mempengaruhi Error

Dalam teknologi satelit navigasi global berbagai penyebab memberi andil dalam memperhitungkan total error sebagai berikut:

1. Satellite clock: walaupun setiap satelit GPS dilengkapi dengan atomic clock yang sangat akurat, error pewaktuan sebesar 10 ns saja cukup menimbulkan error posisi senilai 3 meter.
2. Orbit satelit: Pada umumnya kepastian posisi hanya diketahui sampai perkiraan 1-5 meter.
3. Efek ionosfe: Ionosfer adalah lapisan atmosfer yang ada pada 60-1000 kilometer dari permukaan bumi.
4. Error pada pengukuran waktu tempuh rambat sinyal: receiver GPS hanya mampu mennetukan waktu kedatangan sinyal satelit dengan keakuratan yang terbatas.
5. Multipath: tingkat error bertambah dengan adanya penerimaan sinyal pantulan dari gedung-gedung, pepohonan, pegunungan dan sebagainya.

2.5 Jaringan Seluler

Jaringan seluler digunakan untuk pengiriman dari android ke server jaringan seluler adalah jangkauan umumnya mencakup nasional dengan infrastruktur jaringan *wireless* yang disediakan oleh *wireless service carrier* untuk biaya pemakaian bulanan, mirip dengan langganan ponsel. Untuk transmisi data menggunakan sistem CDMA, GSM, EDGE, 3G, dan HSPDA. Jaringan *wireless* beroperasi dalam sebuah jaringan yang membagi kota atau wilayah kedalam sel-sel yang lebih kecil. Satu sel mencakup beberapa blok kota atau sampai 250 mil persegi. Setiap sel menggunakan sekumpulan frekuensi radio atau saluran-saluran untuk memberikan layanan di area spesifik. Kekuatan radio ini harus dikontrol untuk membatasi jangkauan geografis. Oleh Karena itu, frekuensi yang sama dapat digunakan kembali di sel terdekat. Maka banyak orang dapat melakukan percakapan secara simultan dalam sel yang berbeda di seluruh kota atau wilayah, meskipun mereka berada dalam satu saluran. Dalam setiap sel, terdapat stasiun dasar yang berisi antenna *wireless* dan perlengkapan radio lain. Antenna *wireless* dalam setiap sel akan menghubungkan sumber ke jaringan telepon lokal, internet, ataupun jaringan *wireless* lain.

2.6 Quality of Service

Suatu penilaian terhadap jaringan yang bertujuan untuk mendefinisikan karakteristik dan sifat dari suatu layanan. QoS mengacu pada performansi dari paket-paket IP yang lewat melalui jaringan. Standar yang dijadikan acuan mengenai kualitas layanan tersebut adalah *Telecommunication and Internet Protocol Harmonization Over Network (TIPHON)* dan standar dari Internasional *Telecommunication Union Telecommunication (ITU-T)*. Untuk fungsi pemantauan posisi yang diterima dari GPS maka akan dianalisa adalah *packet loss*, *delay* dan *throughput*.

2.6.1 Delay

Delay merupakan keterlambatan dalam waktu transmisi data dari pengirim dan penerima, satuan dari delay adalah sekon (detik).

2.7 Android

Android adalah sistem operasi berbasis Linux yang dipergunakan sebagai pengelola sumber daya perangkat keras, baik untuk ponsel, smartphone dan juga PC tablet. Secara umum Android adalah platform yang terbuka (Open Source) bagi para pengembang untuk menciptakan aplikasi

mereka sendiri untuk digunakan oleh berbagai piranti bergerak, fungsi dalam tugas akhir ini adalah sebagai pengganti modul GPS. Dimana yang nantinya akan terbaca pada display sistem.

2.7.1 Application Layer

Layer pertama pada OS Android, biasa dinamakan layer Applications. Layer ini biasa digunakan yang berhubungan dengan aplikasi inti yang berjalan pada Android OS. Lapisan aplikasi merupakan lapisan yang paling tampak pada pengguna ketika menjalankan program. Pengguna hanya akan melihat program ketika digunakan tanpa mengetahui proses yang terjadi dibalik lapisan aplikasi. Lapisan ini berjalan dalam Android *runtime* dengan menggunakan kelas dan servis yang tersedia pada *framework* aplikasi. Pada Android semua aplikasi, baik aplikasi inti (*native*) maupun aplikasi pihak ketiga berjalan diatas lapisan aplikasi dengan menggunakan pustaka API (*Application Programming Interface*) yang sama.

2.7.2 Application Framework

Application Framework merupakan layer dimana pembuat aplikasi menggunakan komponen-komponen yang ada di sini untuk membuat aplikasi mereka. Kerangka Aplikasi menyediakan kelas-kelas yang dapat digunakan untuk mengembangkan aplikasi Android. Selain itu juga menyediakan abstraksi generik untuk mengakses perangkat, serta mengatur tampilan *user interface* dan sumber daya aplikasi. Ada beberapa Bagian terpenting dalam kerangka aplikasi Android adalah sebagai berikut:

- *Activity Manager*, berfungsi untuk mengontrol siklus hidup aplikasi dan menjaga keadaan *Backstack* untuk navigasi penggunaan.
- *Content Providers*, berfungsi untuk merangkum data yang memungkinkan digunakan oleh aplikasi lainnya dan seperti daftar nama.
- *Resource Manager*, untuk mengatur sumber daya yang ada dalam program. Serta menyediakan akses sumber daya diluar kode program, seperti karakter, grafik dan file layout.
- *Location Manager*, berfungsi untuk memberikan informasi detail mengenai lokasi perangkat Android berada.

Notification Manager, mencakup berbagai macam peringatan seperti, pesan masuk, dan lain sebagainya yang akan ditampilkan pada *statusbar*.

2.8 Personal Home Page (PHP)

Bahasa pemrograman yang biasa digunakan dalam pembuatan aplikasi berbasis web yang bersifat server-side yang difungsikan untuk membangun suatu website dinamis. PHP menyatu dengan kode HTML, maksudnya adalah beda kondisi, HTML digunakan sebagai pembangun atau pondasi dari kerangka layout web, sedangkan PHP difungsikan sebagai prosesnya sehingga dengan adanya PHP tersebut, web akan sangat mudah di-*maintenance*. PHP berjalan pada sisi server sehingga PHP disebut juga sebagai bahasa *Server Side Scripting*. Artinya bahwa dalam setiap/untuk menjalankan PHP, wajib adanya web server. PHP ini bersifat open source sehingga dapat dipakai secara cuma-cuma dan mampu lintas platform, yaitu dapat berjalan pada sistem operasi Windows maupun Linux. PHP juga dibangun sebagai modul pada web server apache dan sebagai binary yang dapat berjalan sebagai CGI, pada tugas akhir ini peran PHP adalah sebagai Bahasa yang akan digunakan dalam pembuatan display sistem.

2.9 Apache

Merupakan perangkat lunak yang berfungsi menampilkan halaman web yang tepat, sesuai dengan PHP yang telah dibuat. Apache bersifat *open source*, artinya setiap orang boleh menggunakan, mengambil, dan mengubah kode programnya.

2.10 PHPMyAdmin

Merupakan perangkat lunak yang berfungsi untuk mengatur *database* adalah PHPMyAdmin. Dengan PHPMyAdmin *user* dapat dengan mudah membuat tabel, mengisi data, dan masih banyak yang lainnya yang tanpa harus hafal rumus pemrogramannya, namun cukup dengan mengisi tabel-tabel yang telah tersedia.

[Halaman ini sengaja dikosongkan]

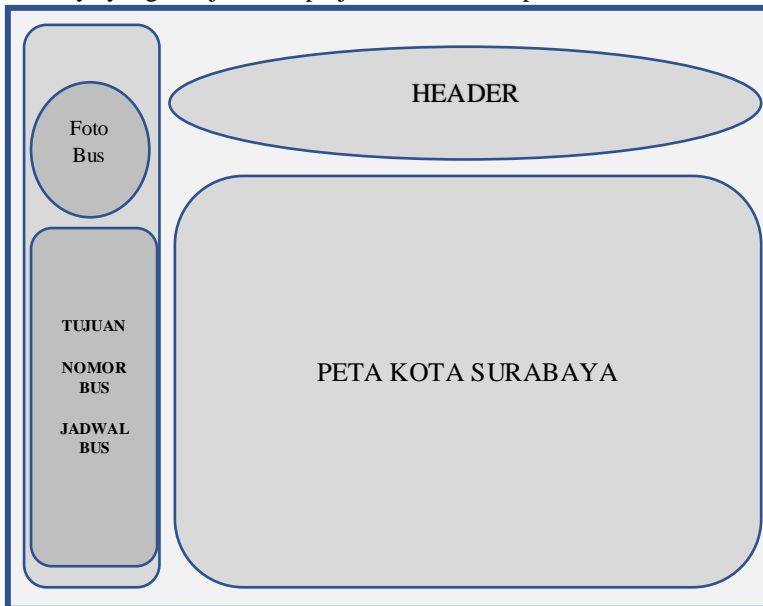
BAB 3

PERANCANGAN

Pada bab ini dijelaskan perancangan sistem pemantauan armada. Proses pertama yang dilakukan yaitu, GPS akan menyalurkan data menuju OBU. Data tersebut berupa waktu dan posisi. Setelah proses telah selesai dilakukan di OBU, kemudian web akan mengambil data yang telah tersimpan di database untuk ditampilkan pada display, sehingga masing-masing armada dapat terpantau pergerakannya. Untuk memudahkan proses perancangan dan implementasi sistem diperlukan alur yang menjelaskan garis besar proses yang dilakukan saat menjalankan tugas akhir.

3.1 Perancangan Tampilan Pemantau Posisi Armada

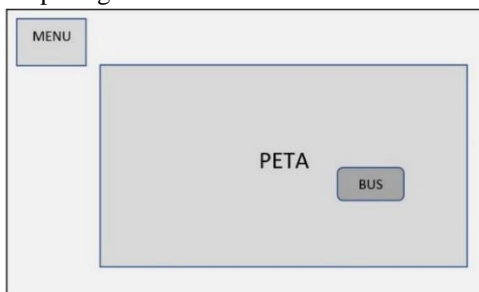
Dalam tahapan ini display yang akan tampil pada ruang kontrol adalah informasi dari tujuan bus, plat nomor bus, foto bus, dan peta kota Surabaya yang menjadi rute perjalanan dari transportasi umum ini



Gambar 3.1 Desain Tampilan Web Pemantau Armada Bus

3.1.1 Rancangan Tampilan Pembaruan Lokasi Armada

Pada tampilan ini akan terlihat perbedaan bus yang bergerak dari satu titik ke titik lain secara lebih jelas, data lokasi dari bus tersebut akan dikirimkan ke server yang kemudian ditampilkan pada display ruang kontrol yang tampilannya akan seperti gambar dibawah ini.



Gambar 3.2 Tampilan Pembaruan Lokasi Armada

3.1.2 Rancangan Informasi Identitas Armada

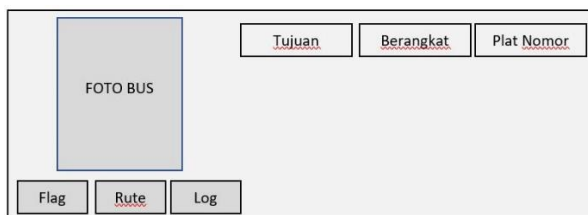
Pada tampilan ini akan muncul informasi mengenai armada yang akan dipantau. Mulai dari foto bus, tujuan, plat nomor dan jadwal keberangkatan dari bus. Semua informasi yang ada mengenai armada disimpan pada database yang telah ditentukan, yang nantinya jika masing-masing dari armada di-klik maka detail informasi yang ada pada database akan muncul pada display di ruang kontrol yang pada saat tugas akhir ini adalah laptop.



Gambar 3.3 Tampilan Informasi Armada

3.1.3 Perancangan Tampilan List Armada

Dimana pada tampilan ini bus akan terlihat lebih detail informasi, mulai dari log perjalanan dan rute yang akan ditempuh. Serta terdapat fungsi untuk menghilangkan bus dari peta untuk sementara dikarenakan bus tidak sedang beroperasi, misal pada hari libur maka otomatis bus yang akan beroperasi di jalanan Surabaya akan dikurangi dibandingkan hari kerja. Tujuan adanya fungsi ini agar bus yang ditampilkan pada display akan fokus pada bus yang beroperasi.



Gambar 3.4 Rancangan Tampilan List Armada

3.2 Perancangan *On-Board Unit (OBU)*

Pada tugas akhir ini OBU hanya mempunyai fungsi GPS dimana yang berfungsi untuk memantau posisi dari armada bus Angkutan masal cepat Surabaya. GPS yang dipakai memanfaatkan HP android, maka dibuatlah aplikasi di dalam android yang dapat mengirimkan data posisi dan waktu untuk dikirimkan ke database. Pengkodean yang akan dilakukan merupakan pengembangan dari google map API yang telah ada, dengan menggabungkan fungsi google map API maka fungsi GPS yang ada pada android akan mengirimkan titik koordinat dari masing-masing armada bus yang tengah beroperasi maupun yang tengah berhenti beroperasi dikarenakan hari libur atau hari tanggal merah. Tahapan yang dilakukan dalam pembuatan Android Package Kit (APK) ini antara lain:

1. Pengkodean menggunakan bahasa pemrograman Java yang dilakukan pada software Android Studio
2. Dilakukan pemrograman untuk mengirimkan data koordinat dari armada bus (HP android) yang kemudian di kirimkan melalui jaringan seluler ke server.
3. Menentukan tampilan untuk APK

4. Memasang APK yang telah jadi pada HP android



Gambar 3.5 Desain Tampilan APK Android

Pada OBU sendiri mempunyai spesifikasi adanya sensor pemantau bahan bakar, sensor GPS, sensor *ticketing*, *public address announcement* dan beberapa sensor lain yang tertanam pada masing-masing armada. Pada tugas akhir ini yang akan dibahas adalah fungsi dari GPS.

3.2.1 Spesifikasi GPS

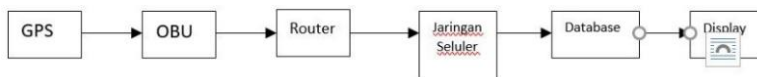
Pada tugas akhir ini menggunakan GPS masyarakat sipil yang tertanam pada android Samsung dimana kepresisian dari GPS masyarakat sipil tersebut dikisaran 3 meter sampai 20 meter dalam kondisi ruangan yang terbuka, dan pada ruangan tertutup tingkat kepresisian GPS dikisaran 4 meter sampai 30 meter, GPS yang digunakan kali ini juga mempunyai sifat GLONASS satelit dimana mempercepat dalam menentukan posisi dari GPS.

3.3 Perancangan Jaringan Komunikasi

Mengacu pada desain manajemen armada kota Surabaya dapat didefinisikan melalui masing-masing bagian yang menjadi pekerjaan yang disusun sebagai tahap penerapan manajemen armada kota Surabaya.

3.3.1 Struktur Jaringan Komunikasi Armada

Alur pengiriman data dari GPS menuju display dapat dilihat pada blok diagram yang akan dijelaskan dibawah, dimana terdapat GPS, OBU, router, jaringan seluler database dan display pada ruang kontrol yang pada tugas akhir ini adalah computer milik pribadi yang telah tersambung pada jaringan internet.



Gambar 3.6 Diagram blok pengiriman data GPS

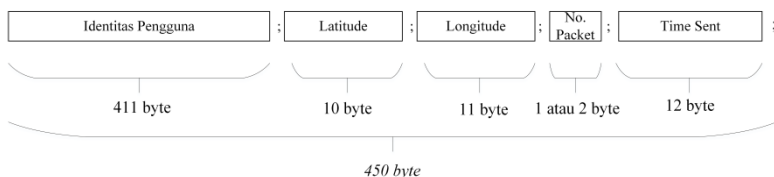
Modul GPS yang telah terintegrasi dengan On-Board Unit akan mengirimkan data berupa data lokasi dari masing-masing armada melalui router yang kemudian akan terkoneksi dengan aplikasi yang telah dirancang untuk memantau setiap armada kemudian data tersebut dikirim melalui jaringan seluler menuju ke database yang telah mempunyai tabel-tabel yang dapat tersinkronisasi untuk diolah dan mengambil informasi mengenai armada kemudian data lokasi tersebut dapat langsung ditampilkan melalui display komputer.

3.4 Alur Perancangan

Perlu dilakukan tahapan perancangan dan implementasi sistem. Perencanaan dibuat untuk memudahkan pekerjaan dari implementasi sistem dimasa yang akan datang, sehingga analisis pada sistem yang telah dibangun dapat dipahami, berikut merupakan diagram alur komunikasi dari pemantau posisi armada

3.4.1 Proses Pengiriman Data GPS ke Jaringan

Di dalam manajemen cerdas ini terdapat perangkat yang disebut On-Board Unit yang didalamnya terdapat modul GPS yang akan mengirimkan ke jaringan berupa data posisi dan waktu. Agar GPS dapat terhubung dengan jaringan maka On-Board Unit harus terkoneksi dengan salah satu jaringan operator. Format paket yang akan dikirimkan seperti gambar 3.7



Gambar 3.7 Format Paket

Dalam proses pengiriman data, data yang dikirimkan akan mengalami proses *encapsulation*. Proses enkapsulasi adalah sebuah proses yang membuat satu jenis paket data jaringan menjadi jenis data lainnya. Enkapsulasi terjadi ketika sebuah protokol yang berada pada lapisan yang lebih rendah menerima data dari protokol yang berada pada lapisan yang lebih tinggi dan meletakkan

data ke format data yang dipahami oleh protokol tersebut. Enkapsulasi terhadap pesan-pesan UDP oleh protokol IP dilakukan dengan menambahkan *header* IP dengan protokol IP nomor 17 (0x11).

3.4.2 Proses Pengiriman Data Jaringan ke Server

Pengiriman berikutnya adalah data GPS dari jaringan ke server. Pada pengiriman ini, data yang tersimpan sementara di OBU akan dikirimkan ke *server*. Proses pengiriman data yang dilakukan menggunakan jaringan seluler karena *On-Board Unit* akan selalu berpindah-pindah tempat.

3.5 Perangkat Server yang Digunakan

Server yang digunakan dalam tugas akhir ini adalah ta01.lawanghosting.pw. Server tersebut merupakan server untuk publikasi alamat *IP-Public speedy*, server ini dianalogikan sebagai *server* yang terdapat pada CC-ROOM. Server tersebut dirancang agar dapat menampung data-data yang dikirimkan oleh GPS yang kemudian pemantauannya akan ditampilkan di layar CC-ROOM yang sementara ini masih berupa *display* pada komputer.

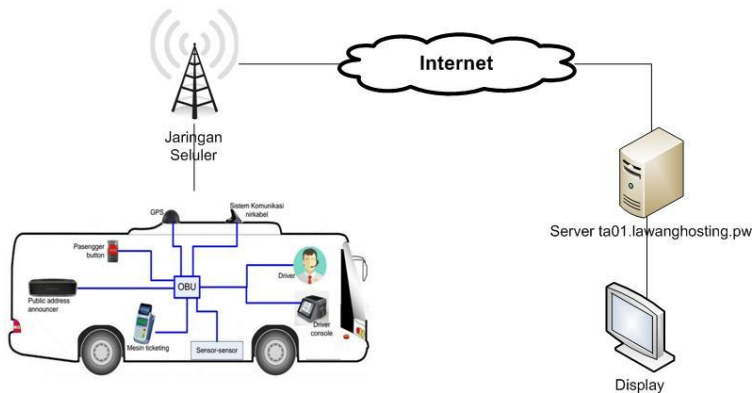
3.6 Perancangan Sistem Pemantau Posisi Armada

Pada perancangan ini dilakukan proses pembuatan sistem yang nantinya akan berada pada CC-ROOM agar dapat memantau posisi armada. Adapun tahapan-tahapan perancangan:

1. Pembuatan arsitektur jaringan pengiriman data GPS
2. Perancangan sistem untuk pengiriman data GPS pada aplikasi berbasis *web*
3. Perancangan database PHPMyAdmin untuk penyimpanan data *longitude*, *latitude*, dan id armada.

3.6.1 Pembuatan Arsitektur Jaringan Pengiriman Data dari GPS

Dalam tahap ini dilakukan pemodelan arsitektur jaringan pengiriman data dari armada ke *server*. Arsitektur jaringan ini digunakan untuk memudahkan saat melakukan pemantauan armada nanti. Berikut arsitektur jaringan pengiriman data GPS yaitu mulai dari OBU yang mempunyai beberapa sensor, tetapi pada tugas akhir ini hanya data dari GPS yang dikirimkan melalui jaringan seluler yang akan disalurkan ke internet kemudian akan menuju ke server database ta01.lawanghosting.pw yang kemudian data tersebut ditampilkan pada display yang ada di ruang kontrol sehingga armada yang beroperasi dapat dipantau sesuai yang diinginkan.

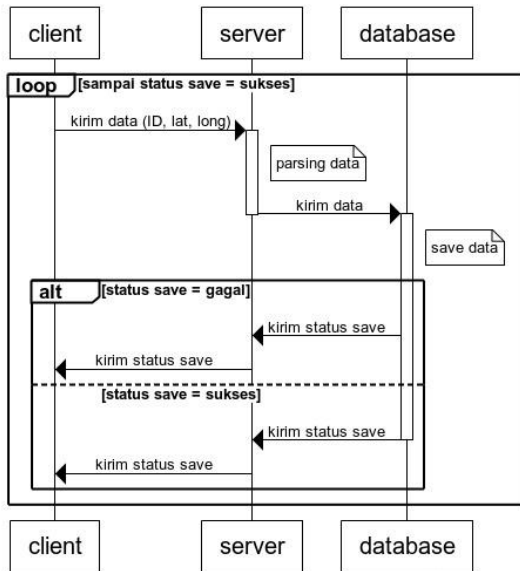


Gambar 3.8 Arsitektur jaringan

Arsitektur dalam Gambar 3.8 adalah arsitektur yang akan digunakan untuk melakukan uji pemantauan posisi. Dalam gambar tersebut terdapat beberapa OBU yang nantinya akan bergerak karena diletakkan pada setiap armada AMC yang terhubung dengan jaringan operator seluler. Kemudian data dari GPS akan dikirim ke *server* yang mempunyai IP Public.

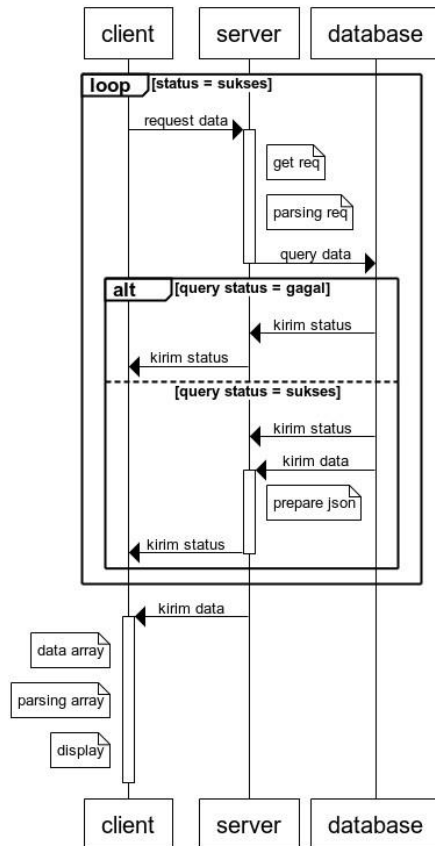
3.6.2 Perancangan Aplikasi Berbasis Web untuk Pemantauan Armada

Aplikasi berbasis web ini dibuat untuk implementasi dari pemantauan setiap armada yang beroperasi yang terintegrasi dengan CC-ROOM dimana dalam situasi ini digunakan laptop sebagai penggantinya. Sebelum aplikasi dibuat diperlukan suatu alur atau diagram protokol aplikasi berbasis web ini. Berikut diagram protokol untuk proses pengiriman data yang akan terjadi pada penelitian yang akan dilakukan, sebagai kunci utama terdapat adanya *client*, *server* dan *database*. Kunci utama tersebut dapat dilihat dari diagram yang ada dibawah ini yaitu alur data yang akan dikirimkan oleh OBU yaitu data posisi yang kemudian melalui server yang telah dibangun dan memanggil semua informasi mengenai armada bus pada database yang telah dibangun dan ditentukan pada tugas akhir ini, sehingga semua informasi dapat dikirimkan ke database dengan baik tanpa adanya kehilangan data dalam perjalanan.



Gambar 3.9 Diagram Protokol Client to Server

Proses pengiriman data dari sisi client dan server juga ditampilkan pada diagram protokol sebagai berikut agar mempermudah dalam pemahaman dari sisi pengambilan data, dimana pengambilan data yang diambil adalah identitas dari bus, dan latitude longitude dari bus tersebut. Setelah itu dikirimkan ke server yang telah dibangun jika belum berhasil dalam pengiriman maka sistem akan otomatis mengirim ulang sampai pengiriman data yang dibutuhkan berhasil dan diterima di *server* dan database.

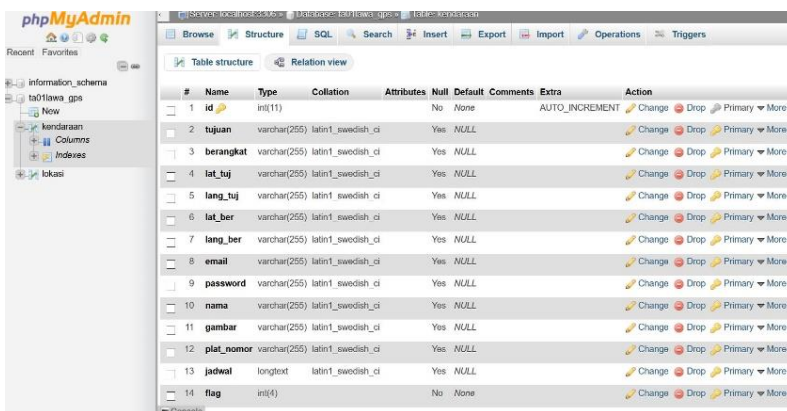


Gambar 3.10 Diagram Protokol Dari Ruang kontrol ke server

Pengambilan data dari sisi *client* juga bisa dijelaskan melalui gambar diatas dimana untuk mengambil data perlu di parsing terlebih dahulu kemudian data akan ditampilkan ke *display* pemantauan. Pada sisi *server* terjadi proses pengambilan data dimana *server* meminta data dari GPS kemudian di parsing untuk dilanjutkan query ke database, jika *query* sukses maka program JSON akan terpanggil dan data dari GPS tersebut berhasil ditampilkan ke display pada ruang kontrol.

3.6.3 Perancangan Database PHPMyAdmin untuk Penyimpanan Data

Server yang digunakan untuk menerima data dalam uji coba ini adalah server ta01.lawanghosting.pw dengan memanfaatkan database PHPMyAdmin pada server. Data yang masuk ke server akan disimpan dalam database ta01law. Database dibuat sesuai dengan format perancangan sistem pemantauan posisi armada, dimana terdapat tabel ta01law_gps. Berikut merupakan gambar yang merupakan tampilan struktur dari tabel ta01law_gps:



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop Primary More
2	tujuan	varchar(255)	latin1_swedish_ci		Yes	NULL			Change Drop Primary More
3	berangkat	varchar(255)	latin1_swedish_ci		Yes	NULL			Change Drop Primary More
4	lat_tuj	varchar(255)	latin1_swedish_ci		Yes	NULL			Change Drop Primary More
5	lang_tuj	varchar(255)	latin1_swedish_ci		Yes	NULL			Change Drop Primary More
6	lat_ber	varchar(255)	latin1_swedish_ci		Yes	NULL			Change Drop Primary More
7	lang_ber	varchar(255)	latin1_swedish_ci		Yes	NULL			Change Drop Primary More
8	email	varchar(255)	latin1_swedish_ci		Yes	NULL			Change Drop Primary More
9	password	varchar(255)	latin1_swedish_ci		Yes	NULL			Change Drop Primary More
10	nama	varchar(255)	latin1_swedish_ci		Yes	NULL			Change Drop Primary More
11	gambar	varchar(255)	latin1_swedish_ci		Yes	NULL			Change Drop Primary More
12	plat_nomor	varchar(255)	latin1_swedish_ci		Yes	NULL			Change Drop Primary More
13	jadwal	longtext	latin1_swedish_ci		Yes	NULL			Change Drop Primary More
14	flag	int(4)			No	None			Change Drop Primary More

Gambar 3.11 Tampilan Struktur dari tabel kendaraan

Gambar 3.10 adalah tampilan tabel yang ada pada database, dalam tabel tersebut terdapat beberapa *field* antara lain:

1. Id, berisi informasi id dari masing-masing armada
2. Tujuan, berisi lokasi yang akan dituju oleh setiap armada
3. Berangkat, berisi informasi lokasi keberangkatan armada
4. Lat_tuj, berisi titik koordinat halte yang akan dituju
5. Lat_ber, berisi titik koordinat keberangkatan armada
6. Lang_tuj, berisi koordinat halte yang akan dituju
7. Lang_tuj, berisi koordinat keberangkatan armada
8. Email, berisi akses login untuk setiap armada
9. Password, berisi akses login untuk setiap armada
10. Nama, berisi informasi kode bus
11. Gambar, berisi foto dari armada yang beroperasi
12. Jadwal, berisi waktu jam beroperasi armada
13. Plat_nomor, berisi identitas plat nomor armada

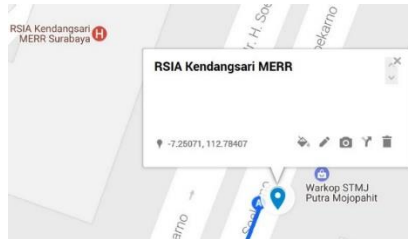
14. Flag, berisi pilihan meng-aktifkan dan me-nonaktifkan armada yang beroperasi pada tampilan display

3.7 Skenario Kinerja

Skenario pada pengujian rancangan ini mempunyai beberapa tahap yang memudahkan dalam pengambilan data yang diperlukan yaitu dilakukan dengan melakukan pembuatan grid pada peta digital antara lain menentukan titik-titik koordinat lokasi yang akan menjadi acuan pengukuran ketepatan GPS dalam menentukan posisi armada kemudian mencatat koordinat teoritis yang telah menjadi acuan pada peta digital. Mendatangi lokasi yang telah ditentukan dalam acuan secara berurutan serta mencatat informasi koordinat dan waktu yang didapat. Mencatat waktu dan posisi yang diterima pada *server* melalui sistem yang telah digunakan. Melalui perbandingan waktu antara data GPS yang dikirimkan oleh android dan data yang diterima oleh *server* dapat ditemukan *delay* yang terjadi pada saat pengiriman data berlangsung pada sistem ini. Melakukan secara berulang kegiatan pengambilan data di titik lokasi yang menjadi acuan, karena GPS tidak akan mencatat titik koordinat yang sama persis sekalipun yang kita jadikan titik pengukuran adalah tempat yang sama. Dari pengukuran tersebut barulah bisa didapatkan tabel yang berisi error terhadap acuan yang ditentukan, *error* terhadap dirinya sendiri, jarak antara server dan titik pengukuran serta deviasi koordinat lokasi.

3.7.1 Pembuatan *Grid* pada Peta Digital

Konsep pembuatan *grid* pada perancangan ini dapat dilihat pada gambar dibawah ini. Pada perancangan ini menggunakan *google maps* sebagai peta *digital*. Grid dilakukan dengan mengambil gambar sepanjang jalan Merak lalu diolah pada *coreldraw* sehingga grid dapat ditentukan untuk penelitian ini kemudian muncul skala pada gambar adalah 3,0163 centimeter untuk setiap 200 meter pada skala aslinya yang mana error sedikit pada peta akan berpengaruh cukup jauh pada kondisi sebenarnya.



Gambar 3.13 Koordinat teoritis di depan RSIA Kendangsari

Lokasi tersebut dipilih berdasarkan titik-titik lokasi yang memungkinkan untuk penandaan secara realita, sehingga tempat pengujian titik koordinat dapat tidak berubah pada realita. Untuk membandingkan tingkat error pada titik acuan.

3.7.3 Peninjauan Lokasi

Bertujuan untuk mendapatkan data lokasi dan waktu yang dikirimkan oleh GPS ke *server* dan membandingkan titik koordinat yang didapatkan dari GPS android yang digunakan dan titik acuan dari grid peta digital yang telah dibuat pada tahap sebelumnya. Lokasi yang didatangi merupakan lokasi sesungguhnya yaitu pada contoh dibawah ini adalah rumah sakit Kendangsari MERR yang mempunyai titik koordinat -7.25071, 112.78407 berdasarkan acuan dari *google maps*.



Gambar 3.14 Peninjauan Lokasi RSIA Kendangsari MERR

Pada tahap ini juga diperlukan pencatatan beberapa data yang dihasilkan oleh GPS android antara lain adalah koordinat yang sesungguhnya pada saat di lokasi peninjauan dan waktu pengiriman data GPS ke server, yaitu di mana posisi GPS yang sebelumnya di non-aktifkan kemudian saat tiba

di lokasi peninjauan langsung diaktifkan, maka secara langsung data dari GPS adalah waktu pengiriman data yang dapat dicatat.

3.7.4 Pembuatan Tabel Perbandingan

Tahapan ini merupakan salah satu yang memudahkan dalam pencatatan data yang telah ada, dimana semua data mengenai koordinat acuan, koordinat titik lokasi peninjauan, error terhadap acuan, rata-rata error, dan koordinat server akan disatukan pada tabel ini sehingga memudahkan dalam pengolahan dan membandingkan data. Contoh tabel yang akan dibuat adalah seperti dibawah ini.

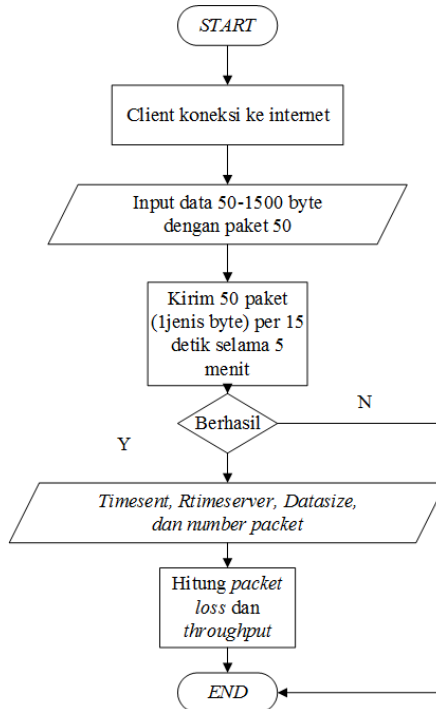
Tabel 3.1 Contoh tabel perbandingan yang akan digunakan

Acuan	RSIA Kendangsari	Error pada acuan	Jarak Server Ke titik 1
-7.25071, 112.78407	-7.25079677,112.78408267	9 meters N (351°)	5.1783 km SW (241.5°)
	-7.25067305,112.78398383	10 meters SE (113°)	5.1753 km SW (241.31°)
	-7.25076689,112.7840052	9 meters NE (48°)	5.1724 km SW (241.42°)
	-7.25078725,112.78398748	12 meters NE (46°)	5.1696 km SW (241.43°)
	-7.25074818,112.78405648	4 meters N (19°)	5.1783 km SW (241.43°)
Rata-Rata	-7.250754428,112.7840231	7 meters NE (46°)	5.1748 km SW (241.42°)

Untuk mengukur jarak pengukuran terhadap server maka diperlukan titik koordinat dari server yang digunakan, yaitu pada gedung Intiland Tower Surabaya, dimana titik koordinatnya adalah -7.273013, 112.742838 data tersebut didapat dari peta digital dengan acuan yang sama dengan titik-titik pengukuran lainnya.

3.8 Skenario Pengujian Jaringan

Pada pengujian ini, akan dilihat QoS jaringan yang telah dirancang. QoS jaringan yang ditinjau adalah *packet loss* dan *throughput*. Langkah-langkah pengujian yang dilakukan yaitu aplikasi telah terkoneksi dengan jaringan Wi-Fi atau jaringan salah satu operator seluler. Kemudian mengatur besar *byte* data sekitar 50-1500 *byte* dan paket yang dikirim sebesar 50 paket yang di-*inputkan* oleh *client* pada aplikasi *testbed* komunikasi terintegrasi AMC. 50 paket ini adalah mewakili 50 OBU. Pengaturan *byte* data tersebut bertujuan untuk mendapatkan rekomendasi berapa ukuran data terbaik yang dapat diterima oleh *server*, data tersebut dikirim melalui jaringan *backbone* dan jaringan seluler. Berikut *flowchart* mengenai skenario pengujian jaringan:



Gambar 0.16 Flowchart pengujian jaringan

Range byte 50-1500 byte tersebut diambil dari standar MTU (*Maximum Transmission Unit*), karena jika ukuran data yang dikirim lebih besar dibandingkan nilai MTU, maka paket yang dikirim lebih besar dibandingkan nilai MTU, paket data yang dikirimkan akan terpecah menjadi beberapa fragmen yang akhirnya tidak jadi terkirim dengan benar. Selanjutnya, data dikirim ke *server*. Pengiriman data ke *server* per 50 paket satu jenis *byte* dengan jeda 15 detik sekali kirim dengan rentang waktu 5 menit. Jika berhasil maka didapatkan *output timesent, Rtimeserver, datasize, dan number packet* yang disimpan dalam *database*. *Timesent, Rtimeserver, datasize, dan number packet* digunakan untuk menghitung QoS, yaitu *packet loss* dan *throughput*.

BAB 4

HASIL DAN ANALISA

4.1 Pendahuluan

Pada bab ini dijelaskan mengenai hasil dari perancangan dan analisa kinerja dari sistem yang telah dibangun menggunakan aplikasi berbasis web. Kemudian menjelaskan hasil error yang didapat dari pengiriman data dari GPS ke server. Terdapat dua data inti dalam hasil analisa ini yaitu data posisi dan waktu.

Selain kedua data tersebut di bab ini juga membahas mengenai situasi di lapangan sesuai dengan pengamatan yang dilakukan, data lapangan antara lain yaitu koordinat posisi GPS saat di lokasi pengujian, ada delapan titik lokasi pengujian yang mendapatkan hasil error rata-rata yang berbeda, terdapat faktor eksternal dan internal yang dapat menyebabkan perbedaan rata-rata error tersebut.

4.2 Realisasi Sistem Pemantau Posisi Armada

Sistem pemantauan ini diwujudkan melalui aplikasi berbasis web dimana dirancanglah suatu web yang dapat menampilkan lokasi dari masing-masing armada yang beroperasi, untuk peta atau jalur dari armada menggunakan goole maps dan titik yang bergerak dimana dalam tugas akhir ini diibaratkan sebagai armada (bus) menggunakan google maps api, karena keterbatasan jumlah titik dalam penggunaan free google maps api maka armada bus yang bisa ditampilkan hanya berjumlah lima armada bus.



Gambar 4.1 Tampilan awal pemantau posisi armada

4.2.1 Pembaruan Lokasi Armada

Cara kerja dari pengiriman data lokasi yang selalu berubah-ubah, pada pengkodean ini digunakan protokol *http post* pada sistem dimana protokol ini mempunyai keuntungan dapat tidak menampilkan lokasi setiap armada bus di URL yang berarti keamanan dari sistem menjadi lebih terjamin.



Gambar 4.2 Tampilan A



Gambar 4.3 Tampilan B

Perbedaan tampilan di display sistem saat berada didepan rumah sakit Kendangsari (gambar A) dan di depan ATM BNI kampus C Unair (gambar B).

4.2.2 Informasi Identitas Armada

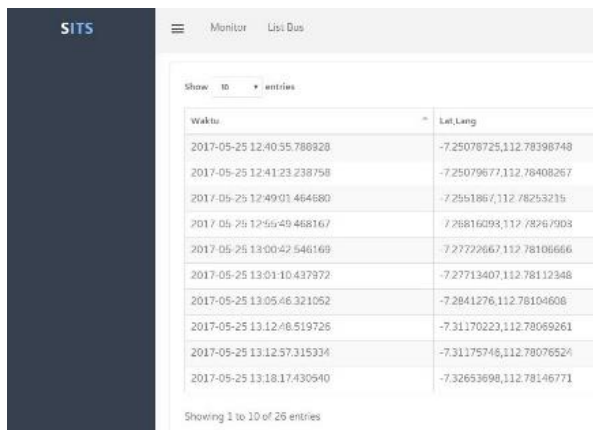
Beberapa informasi yang ditampilkan pada fungsi sistem ini antara lain mengenai foto dari armada bus, tujuan, keberangkatan dan jadwal.



Gambar 4.4 Tampilan identitas armada

4.2.3 Tampilan Waktu dan Koordinat Saat Data Diterima

Pada tampilan ini data dari database ditampilkan secara realtime untuk mengetahui kapan data dari GPS diterima oleh server, sehingga jika terjadi delay yang cukup lama akan tercatat oleh database dari sistem. Berikut tampilannya:



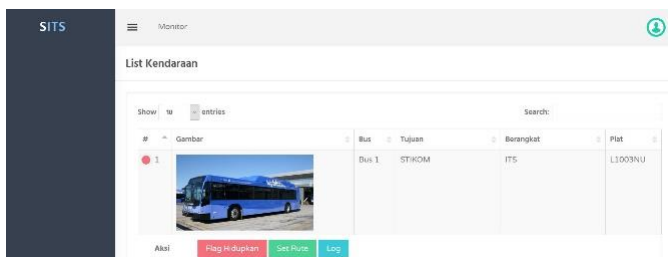
The screenshot shows the SITS Monitor interface. It features a sidebar with the SITS logo and a main content area with a 'Monitor' tab. Below the tab, there's a 'Show: 10 entries' dropdown. The main table displays a list of data entries with columns for 'Waktu' (Time) and 'Lat, Long' (Latitude, Longitude). The table contains 10 rows of data, showing timestamps and coordinates. At the bottom, it says 'Showing 1 to 10 of 26 entries'.

Waktu	Lat, Long
2017-05-25 12:40:55.788928	-7.25078725,112.78398748
2017-05-25 12:41:23.238758	-7.25079677,112.78408267
2017-05-25 12:49:01.464680	-7.2551867,112.78253235
2017-05-25 12:50:49.468167	-7.26816293,112.78267903
2017-05-25 13:00:42.546169	-7.27722667,112.78106666
2017-05-25 13:01:10.437972	-7.27713407,112.78112348
2017-05-25 13:05:46.321052	-7.2841276,112.78104608
2017-05-25 13:12:46.519726	-7.31170223,112.78069261
2017-05-25 13:12:57.315334	-7.31175746,112.78076524
2017-05-25 13:18:17.430940	-7.32653698,112.78146771


Gambar 4.5 Tampilan waktu penerimaan data

4.2.4 Hasil Tampilan List Armada

Disini terdapat menu untuk memilih bus mana saja yang akan ditampilkan pada display, dengan mengklik tombol flag maka otomatis bus akan menghilang dari peta yang ada pada tampilan. Fungsi ini dibuat dengan tujuan jika pada hari libur bus yang beroperasi tidak akan sebanyak pada hari biasa yang terdapat orang kantor dan anak sekolah.



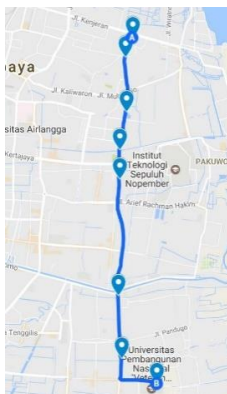
The screenshot shows the SITS Monitor interface with a 'List Kendaraan' (List of Vehicles) section. It includes a search bar and a table with columns for '#', 'Gambar' (Image), 'Bus', 'Tujuan' (Destination), 'Berangkat' (Departure), and 'Plat' (Plate). The table lists one vehicle, 'Bus 1', with destination 'STIKOM', departure 'ITS', and plate 'L1003NU'. Below the table, there are buttons for 'Aksi' (Action), 'Flag Hidden', 'Get Route', and 'Log'.

#	Gambar	Bus	Tujuan	Berangkat	Plat
1		Bus 1	STIKOM	ITS	L1003NU

Gambar 4.6 Tampilan List Armada yang akan dipantau

4.2.4 Hasil Pembuatan Titik Acuan

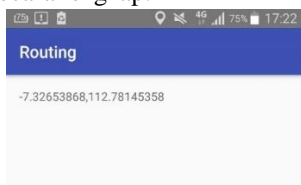
Untuk mempermudah dalam penentuan titik lokasi uji maka dibuatlah acuan pada google map dengan delapan titik di sepanjang jalan MERR berikut adalah hasilnya:



Gambar 4.7 Titik acuan

4.3 Hasil Tampilan Perancangan On-Board Unit

Pembuatan dilakukan menggunakan android studio yang menggunakan bahasa pemrograman *java script* untuk menampilkan koordinat lokasi dari armada yang tengah beroperasi, fungsi pada On-Board Unit ini hanya untuk mengirimkan titik koordinat dan waktu terima pada databse untuk pemantauan posisi. Tampilan apk yang dibangun sebagai berikut, ketika aplikasi android tersebut dibuka maka secara otomatis GPS dari android tersebut akan mengirimkan data lokasi ke database dan mempeprbarui lokasi pada display di ruang kontrol, sehingga posisi lokasi dari setiap armada dapat tercatat pada database secara lengkap.



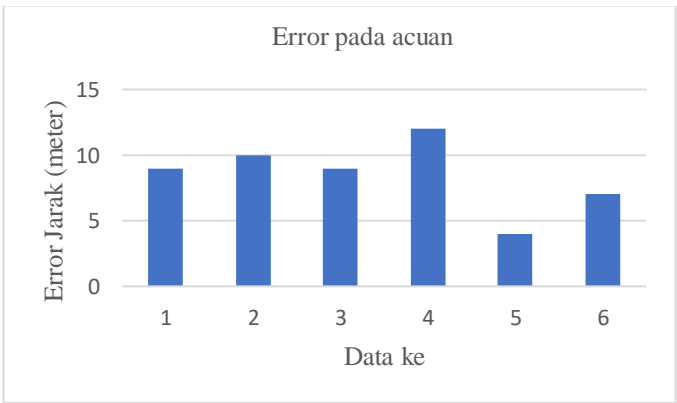
Gambar 4.8 Tampilan pada Android sebagai fungsi GPS

4.4 Evaluasi Error Posisi

Dilakukannya pengukuran pertama yang berlokasi pada titik koordinat -7.25071, 112.78407 jika mengacu pada google maps, yaitu di rumah sakit Kendangsari dimana bisa dilihat pada tabel. Kemudian hasil pengukuran yang tercatat pada server mempunyai rata-rata koordinat -7.250754428,112.7840231 untuk rata-rata error pada hasil pengukuran ini bisa mencapai 7 meter dan jarak rata-rata antara server ke titik pengukuran sekitar 5,1748 km.

Tabel 4.1 Hasil Pengukuran pada lokasi rumah sakit kendangsari

Acuan	Lokasi Ukur	Error	Jarak Server Ke titik 1
-7.25071, 112.78407	-7.25079677,112.78408267	9 meters N (351°)	5.1783 km SW (241.5°)
-7.25071, 112.78407	-7.25067305,112.78398383	10 meters SE (113°)	5.1753 km SW (241.31°)
-7.25071, 112.78407	-7.25076689,112.7840052	9 meters NE (48°)	5.1724 km SW (241.42°)
-7.25071, 112.78407	-7.25078725,112.78398748	12 meters NE (46°)	5.1696 km SW (241.43°)
-7.25071, 112.78407	-7.25074818,112.78405648	4 meters N (19°)	5.1783 km SW (241.43°)
Rata-Rata	-7.250754428,112.7840231	7 meters NE (46°)	5.1748 km SW (241.42°)



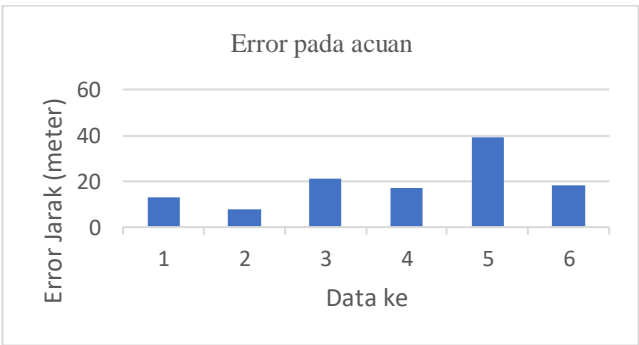
Gambar 4.9 Grafik Error terhadap acuan titik 1

Dari data yang telah ada dapat dianalisa bahwa faktor dari errornya posisi dari armada yang ada adalah faktor keandalan dari GPS yang digunakan, faktor menjauhnya posisi dari server tidaklah berpengaruh pada pengujian ini.

Untuk pengukuran kedua yang berlokasi pada titik koordinat - 7.2553, 112.78242 jika mengacu pada google maps, yaitu di SPBU MERR dimana bisa dilihat pada tabel. Kemudian hasil pengukuran yang telah tercatat pada server yang ada mempunyai rata-rata koordinat - 7.255161339,112.7825153 untuk rata-rata error pada hasil pengukuran ini bisa mencapai 18 meter dan jarak rata-rata antara server ke titik pengukuran sekitar 4,807km.

Tabel 4.2 Hasil Pengukuran pada lokasi rumah sakit kendangsari

Acuan	Lokasi Ukur	Error	Jarak Server Ke Titik 2
- 7.2553, 112.78242	-7.255298,112.7825448	13 meters W (269°)	4.8037 km SW (245.78°)
- 7.2553, 112.78242	-7.25523322,112.78245175	8 meters SW (205°)	4.7973 km SW (245.66°)
- 7.2553, 112.78242	- 7.255111935,112.78248031	21 meters S (197°)	4.8057 km SW (245.52°)
- 7.2553, 112.78242	-7.2551867,112.78253215	17 meters SW (224°)	4.8075 km SW (245.64°)
- 7.2553, 112.78242	-7.25497684,112.78256771	39 meters SW (204°)	4.8208 km SW (245.41°)
Rata-Rata Koordinat	-7.255161339,112.7825153	18 meters SW (214°)	4.807 km SW (245.6°)

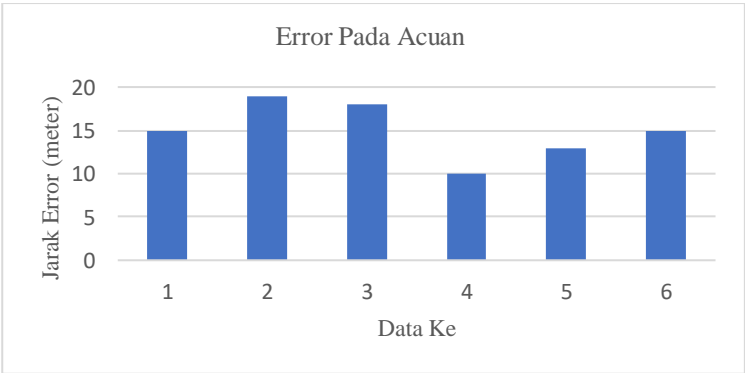


Gambar 4.10 Grafik Error terhadap acuan titik 2

Untuk pengukuran ketiga yang berlokasi pada titik koordinat - 7.26817, 112.78258 jika mengacu pada google maps, yaitu di SPBU MERR dimana bisa dilihat pada tabel. Kemudian hasil pengukuran yang telah tercatat pada server yang ada mempunyai rata-rata koordinat -- 7.268132606,112.7827165 untuk rata-rata error pada hasil pengukuran ini bisa mencapai 18 meter dan jarak rata-rata antara server ke titik pengukuran sekitar 4,42 km.

Tabel 4.3 Hasil Pengukuran pada lokasi ATM BNI Unair Kampus C

Acuan	Lokasi Ukur	Error	Jarak Server ke Titik 3
-7.26817, 112.78258	-7.26812848 ,112.78114427	15 meters W (252°)	4.2612 km W (262.68°)
-7.26817, 112.78258	-7.26809431, 112.7827419	19 meters SW (244°)	4.4365 km W (262.92°)
-7.26817, 112.78258	- 7.26815301,112.78275101	18 meters W (264°)	4.4367 km W (263°)
-7.26817, 112.78258	- 7.26816093,112.78267903	10 meters W (264°)	4.4287 km W (263°)
-7.26817, 112.78258	-7.2681263,112.78269888	13 meters W (249°)	4.4314 km W (262.95°)
Rata-Rata Koordinat	- 7.268132606,112.7827165	15 meters W (254°)	4.42 kilometers W (263°)

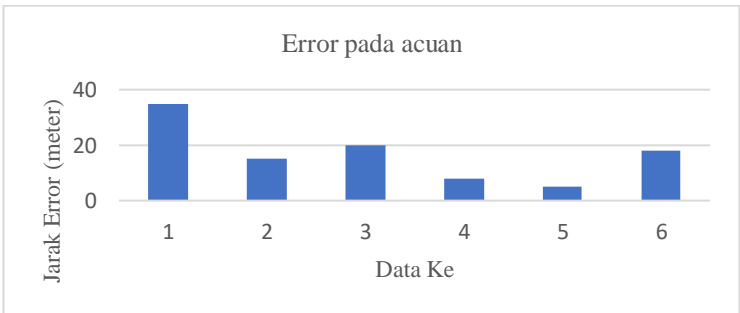


Gambar 4.11 Grafik Error terhadap acuan titik 3

Untuk pengukuran keempat yang berlokasi pada titik koordinat - 7.27718, 112.78104 jika mengacu pada google maps, yaitu di ATM BNI Unair C dimana bisa dilihat pada tabel. Kemudian hasil pengukuran yang telah tercatat pada server yang ada mempunyai rata-rata koordinat - 7.277110964,112.7811286 untuk rata-rata error pada hasil pengukuran ini bisa mencapai 12 meter dan jarak rata-rata antara server ke titik pengukuran sekitar 4,24 km.

Tabel 4.4 Hasil Pengukuran pada lokasi Galaxy Mall

Acuan	Lokasi Ukur	Error	Jarak server ke titik 4
-7.27718, 112.78104	-7.27687459,112.78114427	35 meters S (198°)	4.2481 km W (275°)
-7.27718, 112.78104	-7.27705614,112.78111345	15 meters SW (210°)	4.2468 km W (276.08)
-7.27718, 112.78104	-7.27716175,112.78122372	20 meters W (264°)	4.2468 km W (276.08)
-7.27718, 112.78104	-7.27722667,112.78106666	5 meters NW (330°)	4.2438 km W (276.34)
-7.27718, 112.78104	-7.27723567,112.78109503	8 meters NW (315°)	4.247 km W (276.35)



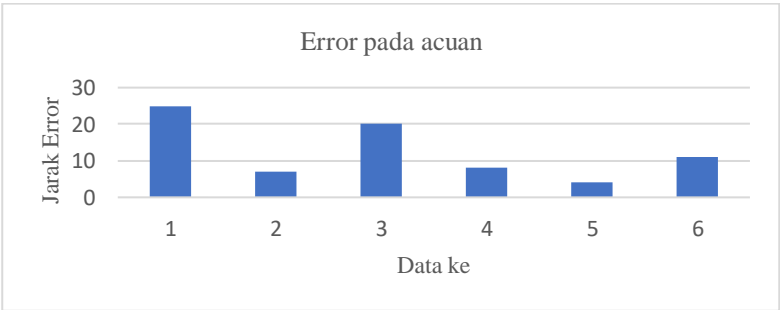
Gambar 4.12 Grafik Error terhadap acuan titik 4

Untuk pengukuran kelima yang berlokasi pada titik koordinat -7.28413, 112.78098 jika mengacu pada google maps, yaitu di Hartono Elektronik MERR dimana bisa dilihat pada tabel. Kemudian hasil pengukuran yang telah tercatat pada server yang ada mempunyai rata-rata koordinat -7.284131496,112.7810825 untuk rata-rata error pada hasil

pengukuran ini bisa mencapai 11 meter dan jarak rata-rata antara server ke titik pengukuran sekitar 4.39 km.

Tabel 4.5 Hasil Pengukuran pada lokasi Hartono Elektronik

Acuan	Lokasi Ukur	Error	Jarak server ke titik 5
-7.28413, 112.78098	-7.2840464,112.78119276	25 meters W (248°)	4.406 km W (286°)
-7.28413, 112.78098	-7.2841276,112.78104608	7 meters W (267°)	4.393 km W (286.34)
-7.28413, 112.78098	-7.28426387,112.78110679	20 meters NW (316°)	4.4037 km W (286.51)
-7.28413, 112.78098	-7.2841209,112.78105494	8 meters W (263°)	4.3938 km W (286.63)
-7.28413, 112.78098	-7.28409871,112.78101194	4 meters SW (225°)	4.3885 km W (286.32)
Rata-Rata Koordinat	-7.284131496,112.7810825	11 meters W (270°)	4.39 kilometers W (286°)

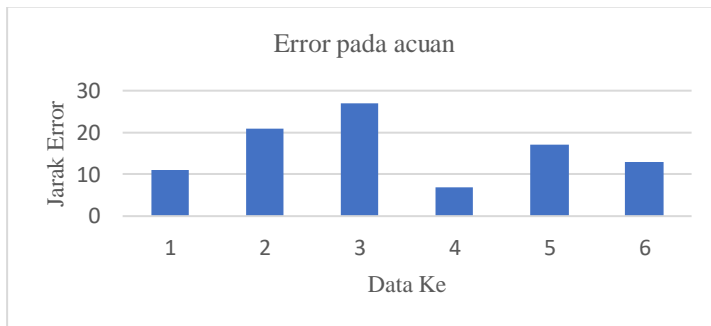


Gambar 4.13 Grafik Error terhadap acuan titik 5

Untuk pengukuran keenam yang berlokasi pada titik koordinat -7.31164, 112.78071 jika mengacu pada google maps, yaitu di Kampus Stikom dimana bisa dilihat pada tabel. Kemudian hasil pengukuran yang telah tercatat pada server yang ada mempunyai rata-rata koordinat -7.311747966,112.7807572 untuk rata-rata error pada hasil pengukuran ini bisa mencapai meter dan jarak rata-rata antara server ke titik pengukuran sekitar 5.98 km.

Tabel 4.6 Hasil Pengukuran pada lokasi Kampus Stikom

Acuan	Lokasi Ukur	Error	Jarak Server ke titik 6
-7.31164, 112.78071	-7.31168619,112.78080537	11 meters NW (295°)	6.0041 km NW (315.76°)
-7.31164, 112.78071	-7.31181586,112.78079263	21 meters NW (334°)	6.0134 km NW (315.87°)
-7.31164, 112.78071	-7.31179604,112.78090968	27 meters NW (308°)	6.0208 km NW (315.76°)
-7.31164, 112.78071	-7.31170223,112.78069261	7 meters N (15°)	5.9967 km NW (315.86°)
-7.31164, 112.78071	-7.31173951,112.78058573	17 meters NE (51°)	5.9914 km NW (315.97°)
Rata-rata Koordinat	-7.311747966,112.7807572	13 meters NW (336°)	5.98 kilometers NW (315°)

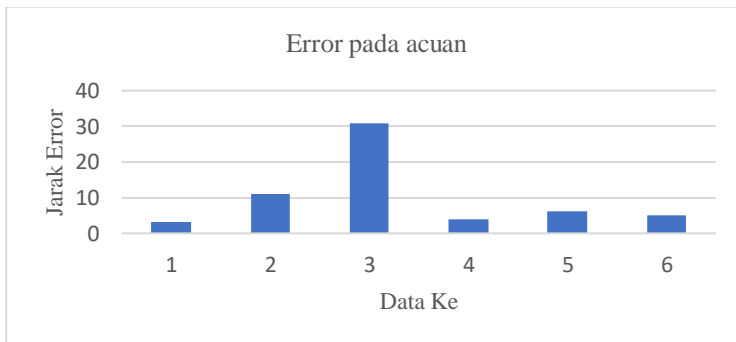


Gambar 4.14 Grafik Error terhadap acuan titik 6

Untuk pengukuran ketujuh yang berlokasi pada titik koordinat -7.32652, 112.78143 jika mengacu pada google maps, yaitu di Kampus Stikom dimana bisa dilihat pada tabel. Kemudian hasil pengukuran yang telah tercatat pada server yang ada mempunyai rata-rata koordinat - 7.326565548,112.7814522 untuk rata-rata error pada hasil pengukuran ini bisa mencapai 5 meter dan jarak rata-rata antara server ke titik pengukuran sekitar 7.29 km.

Tabel 4.7 Hasil Pengukuran pada lokasi Bebek Harissa

Acuan	Lokasi Ukur	Error	Jarak server ke titik 7
-7.32652, 112.78143	-7.32653868,112.78145358	3 meters NW (308°)	7.3208 km NW (324.41)
-7.32652, 112.78143	-7.32642141,112.78145121	11 meters S (192°)	7.31 km NW (324.35)
-7.32652, 112.78143	-7.3268026,112.78140287	31 meters N (5°)	7.3414 km NW (324.58)
-7.32652, 112.78143	-7.32653698,112.78146771	4 meters NW (294°)	7.3215 km NW (324.4)
-7.32652, 112.78143	-7.32652807,112.78148584	6 meters W (278°)	7.3219 km NW (324.38)
Rata-rata Koordinat	-7.326565548,112.7814522	5 meters NW (334°)	7.29 kilometers NW (324°)



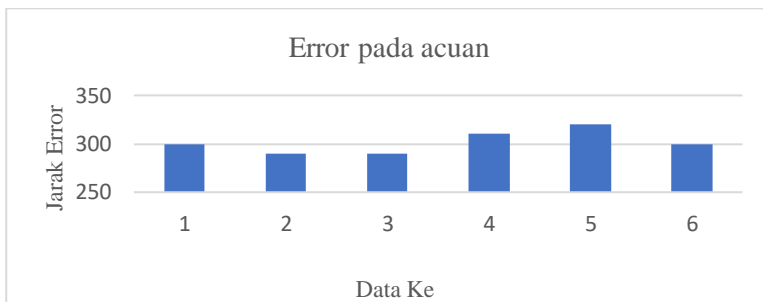
Gambar 4.15 Grafik Error terhadap acuan titik 7

Untuk pengukuran terakhir yang berlokasi pada titik koordinat -7.33256, 112.78983 jika mengacu pada google maps, yaitu di BNI kampus UPN dimana bisa dilihat pada tabel. Kemudian hasil pengukuran yang telah tercatat pada server yang ada mempunyai rata-rata koordinat - 7.331902726,112.7871564 untuk rata-rata error pada hasil pengukuran ini bisa mencapai 0.30km meter dan jarak rata-rata antara server ke titik pengukuran sekitar 8.38 km.

Tabel 4.8 Hasil Pengukuran pada lokasi BNI Kampus UPN

Acuan	Lokasi Ukur	Error	Jarak server ke titik 8
-------	-------------	-------	-------------------------

-7.33256, 112.78983	-7.33187216,112.78717754	0.30 kilometers E (104°)	8.1724 km NW (323.23°)
-7.33256, 112.78983	-7.33183897,112.78726302	0.29 kilometers E (105°)	8.1751 km NW (323.16°)
-7.33256, 112.78983	-7.33194419,112.78724282	0.29 kilometers E (103°)	8.1831 km NW (323.23°)
-7.33256, 112.78983	-7.33193408,112.78710805	0.31 kilometers E (102°)	8.1733 km 323.3 (NW°)
-7.33256, 112.78983	-7.33192423,112.78699047	0.32 kilometers E (102°)	8.1647 km NW (323.37°)
Rata-rata Koordinat	-7.331902726,112.7871564	0.30 kilometers E (103°)	8.1737 km NW (321°)



Gambar 4.16 Grafik Error terhadap acuan titik 8

Tabel 4.9 Data Pengukuran Gabungan

Acuan	Lokasi Ukur	Error	Jarak Server Ke titik Pengukuran
-7.25071, 112.78407	-7.25079677,112.78408267	9 meters N (351°)	5.1783 km SW (241.5°)
-7.25071, 112.78407	-7.25067305,112.78398383	10 meters SE (113°)	5.1753 km SW (241.31°)
-7.25071, 112.78407	-7.25076689,112.7840052	9 meters NE (48°)	5.1724 km SW (241.42°)
-7.25071, 112.78407	-7.25078725,112.78398748	12 meters NE (46°)	5.1696 km SW (241.43°)
-7.25071, 112.78407	-7.25074818,112.78405648	4 meters N (19°)	5.1783 km SW (241.43°)
Tabel 4.9 Lanjutan			
-7.25071, 112.78407	-7.250754428,112.7840231	7 meters NE (46°)	5.1748 km SW (241.42°)
- 7.2553, 112.78242	-7.255298,112.7825448	13 meters W (269°)	4.8037 km SW (245.78°)

- 7.2553, 112.78242	-7.25523322,112.78245175	8 meters SW (205°)	4.7973 km SW (245.66°)
- 7.2553, 112.78242	-7.255111935,112.78248031	21 meters S (197°)	4.8057 km SW (245.52°)
- 7.2553, 112.78242	-7.2551867,112.78253215	17 meters SW (224°)	4.8075 km SW (245.64°)
- 7.2553, 112.78242	-7.25497684,112.78256771	39 meters SW (204°)	4.8208 km SW (245.41°)
- 7.2553, 112.78242	-7.255161339,112.7825153	18 meters SW (214°)	4.807 km SW (245.6°)
-7.26817, 112.78258	-7.26812848 ,112.78114427	15 meters W (252°)	4.2612 km W (262.68°)
-7.26817, 112.78258	-7.26809431, 112.7827419	19 meters SW (244°)	4.4365 km W (262.92°)
-7.26817, 112.78258	-7.26815301,112.78275101	18 meters W (264°)	4.4367 km W (263°)
-7.26817, 112.78258	-7.26816093,112.78267903	10 meters W (264°)	4.4287 km W (263°)
-7.26817, 112.78258	-7.2681263,112.78269888	13 meters W (249°)	4.4314 km W (262.95°)
-7.26817, 112.78258	-7.268132606,112.7827165	15 meters W (254°)	4.42 kilometers W (263°)
-7.27718, 112.78104	-7.27687459,112.78114427	35 meters S (198°)	4.2481 km W (275°)
-7.27718, 112.78104	-7.27705614,112.78111345	15 meters SW (210°)	4.2468 km W (276.08)
-7.27718, 112.78104	-7.27716175,112.78122372	20 meters W (264°)	4.2468 km W (276.08)
-7.27718, 112.78104	-7.27722667,112.78106666	5 meters NW (330°)	4.2438 km W (276.34)
-7.27718, 112.78104	-7.27723567,112.78109503	8 meters NW (315°)	4.247 km W (276.35)
-7.27718, 112.78104	-7.277110964,112.7811286	12 meters SW (232°)	4.24 kilometers W (276°)
-7.28413, 112.78098	-7.2840464,112.78119276	25 meters W (248°)	4.406 km W (286°)
-7.28413, 112.78098	-7.2841276,112.78104608	7 meters W (267°)	4.393 km W (286.34)
-7.28413, 112.78098	-7.28426387,112.78110679	20 meters NW (316°)	4.4037 km W (286.51)
Tabel 4.9 Lanjutan			
7.28413, 112.78098	-7.2841209,112.78105494	8 meters W (263°)	4.3938 km W (286.63)
-7.28413, 112.78098	-7.28409871,112.78101194	4 meters SW (225°)	4.3885 km W (286.32)
-7.28413, 112.78098	-7.284131496,112.7810825	11 meters W (270°)	4.39 kilometers W (286°)

-7.31164, 112.78071	-7.31168619,112.78080537	11 meters NW (295°)	6.0041 km NW (315.76°)
-7.31164, 112.78071	-7.31181586,112.78079263	21 meters NW (334°)	6.0134 km NW (315.87°)
-7.31164, 112.78071	-7.31179604,112.78090968	27 meters NW (308°)	6.0208 km NW (315.76°)
-7.31164, 112.78071	-7.31170223,112.78069261	7 meters N (15°)	5.9967 km NW (315.86°)
-7.31164, 112.78071	-7.31173951,112.78058573	17 meters NE (51°)	5.9914 km NW (315.97°)
-7.31164, 112.78071	-7.311747966,112.7807572	13 meters NW (336°)	5.98 kilometers NW (315°)
-7.32652, 112.78143	-7.32653868,112.78145358	3 meters NW (308°)	7.3208 km NW (324.41)
-7.32652, 112.78143	-7.32642141,112.78145121	11 meters S (192°)	7.31 km NW (324.35)
-7.32652, 112.78143	-7.3268026,112.78140287	31 meters N (5°)	7.3414 km NW (324.58)
-7.32652, 112.78143	-7.32653698,112.78146771	4 meters NW (294°)	7.3215 km NW (324.4)
-7.32652, 112.78143	-7.32652807,112.78148584	6 meters W (278°)	7.3219 km NW (324.38)
-7.32652, 112.78143	-7.326565548,112.7814522	5 meters NW (334°)	7.29 kilometers NW (324°)
-7.33256, 112.78983	-7.33187216,112.78717754	0.30 kilometers E (104°)	8.1724 km NW (323.23°)
-7.33256, 112.78983	-7.33183897,112.78726302	0.29 kilometers E (105°)	8.1751 km NW (323.16°)
-7.33256, 112.78983	-7.33194419,112.78724282	0.29 kilometers E (103°)	8.1831 km NW (323.23°)
-7.33256, 112.78983	-7.33193408,112.78710805	0.31 kilometers E (102°)	8.1733 km 323.3 (NW°)
-7.33256, 112.78983	-7.33192423,112.78699047	0.32 kilometers E (102°)	8.1647 km NW (323.37°)
-7.33256, 112.78983	-7.331902726,112.7871564	0.30 kilometers E (103°)	8.1737 km NW (321°)

Dari data yang telah ditampilkan diatas maka bisa didapatkan rata-rata error keseluruhan yang akan menjadi toleransi dari armada yang ditampilkan pada display di ruang kontrol adalah sakitar 14 meter.



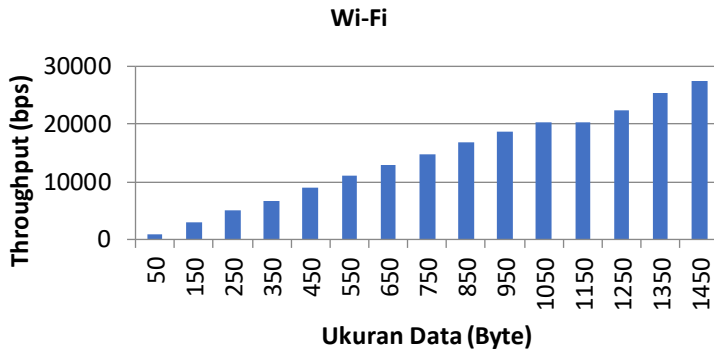
Pada gambar diatas menunjukkan posisi yang berbeda dari lokasi pengukuran dan lokasi yang terbaca pada



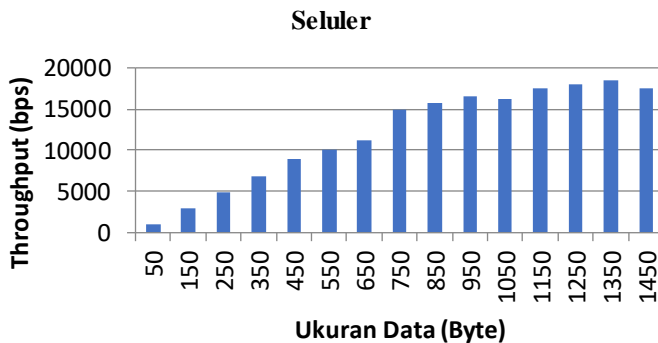
d **Gambar 4.17** Tampilan di **Gambar 4.18** Tampilan yang akan p lokasi pengukuran a c terbaca pada display yang berbeda akan maka pengamat display akan mengira bus berada di lajur lainnya karena error yang dapat terjadi sekitar 14 meter dari posisi yang sebenarnya.

4.5 Pengujian dan Analisis *Troughput*

Perhitungan parameter ini akan digunakan untuk mendapatkan berapa *byte* atau jumlah panjang paket data yang optimal untuk dikirimkan pada *server*. Pengujian ini dilakukan dengan mengirimkan data GPS 50 paket secara terus menerus selama 5 menit dengan jeda 15 detik per 50 paket. Ukuran data yang dikirim ke *server* sebesar 50-1500 *byte*. Pengujian ini dilakukan dengan menggunakan dua laptop yang akan mengirim data secara bersamaan ke *server*. Satu laptop terkoneksi dengan jaringan Wi-Fi dan satu laptop terkoneksi dengan jaringan seluler. Pengujian ini menghasilkan perbandingan antara ukuran data (*byte*) yang dikirim dengan *throughput* ketika pengiriman data menggunakan jaringan Wi-Fi dan jaringan seluler. Berikut hasil dari uji coba :



Gambar 0.19 *Throughput* pada Wi-Fi



Gambar 0.20 *Throughput* pada Jaringan Seluler

Pengujian ini menghasilkan nilai *throughput* ketika pengiriman data melalui jaringan Wi-Fi dan jaringan seluler. Pada jaringan Wi-Fi didapatkan hasil *throughput* paling besar terletak pada ukuran data yang paling besar dan yang terkecil terletak pada ukuran data yang paling kecil. Ukuran data terbesar yaitu 1450 *byte* dengan nilai *throughput* sebesar 27507.44 bps dan ukuran data terkecil yaitu 50 *byte* dengan nilai *throughput* sebesar 962.0533bps. Pada jaringan seluler didapatkan hasil *throughput* paling besar terletak pada ukuran data sebesar 1350 *byte* dengan nilai *throughput* sebesar 18406.77bps. Sedangkan *throughput* terkecil ketika pengiriman data melalui

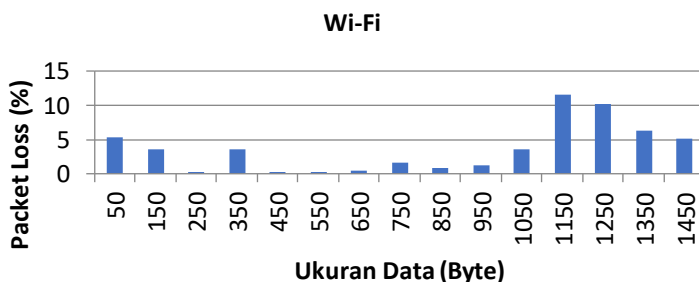
jaringan seluler terletak pada ukuran data paling kecil. Ukuran data paling kecil yaitu 50 byte dengan nilai *throughput* sebesar 963.52bps.

Dengan hasil pengujian tersebut, dapat dianalisis jika ukuran data semakin besar, maka nilai *throughput* yang dihasilkan semakin besar. Nilai *throughput* didapatkan dari pembagian antara jumlah ukuran data yang diterima dengan waktu pengiriman ke *server*. Namun pada jaringan seluler nilai *throughput* terbesar tidak terletak pada ukuran data tertinggi, hal ini terjadi karena pada 1450 byte banyak paket yang hilang.

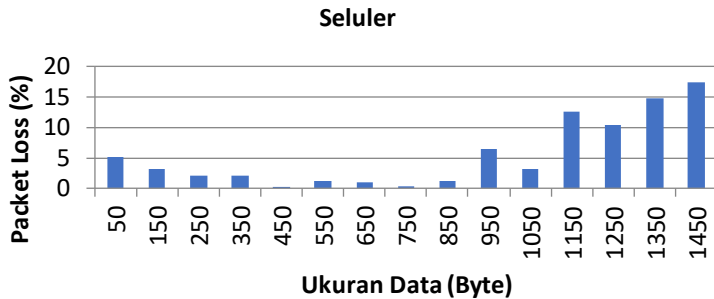
4.6 Pengujian dan Analisa *Packet Loss*

Pengujian *packet loss* digunakan untuk mengetahui paket data yang hilang sewaktu dilakukan pengiriman data dari *client* ke *server*. Pengujian ini dilakukan dengan mengirimkan data 50 paket secara terus-menerus selama 5 menit dengan jeda 15 detik per paket. Ukuran data yang dikirim ke *server* sebesar 50-1500 byte. Pengujian ini dilakukan dengan menggunakan dua laptop yang akan mengirim data secara bersamaan ke *server*. Satu laptop terkoneksi dengan jaringan Wi-Fi dan satu laptop terkoneksi dengan jaringan seluler. Hasil dari pengujian ini terdapat di tabel yang dilampirkan.

Pengujian ini menghasilkan perbandingan antara ukuran data (*byte*) yang dikirim dengan *packet loss* ketika pengiriman data menggunakan jaringan Wi-Fi dan jaringan seluler. Berikut hasil dari uji coba :



Gambar 0.21 Grafik *packet loss* pada Wi-Fi



Gambar 0.22 Grafik *packet loss* pada seluler

Pengujian ini menghasilkan nilai *packet loss* terbesar dan terkecil pada pengiriman data melalui jaringan Wi-Fi dan jaringan seluler. Pada jaringan Wi-Fi, *packet loss* terbesar dengan nilai sebesar 11.47% terletak pada ukuran data 1150 *byte*. *Packet loss* terkecil dengan nilai sebesar 0.267% pada ukuran data 250, 450, dan 550 *byte*. Pada jaringan seluler, *packet loss* terbesar dengan nilai sebesar 17.45% terletak pada ukuran data 1450 *byte*. Sedangkan untuk nilai *packet loss* terkecil ketika pengiriman data melalui jaringan seluler sebesar 0.13% terletak pada ukuran data 450 *byte*.

Berdasarkan grafik hasil pengujian, dapat dianalisis bahwa timbulnya *packet loss* disebabkan oleh kepadatan trafik yang menimbulkan antrian sehingga beberapa paket data tidak diterima oleh server. *Packet loss* seluler lebih besar daripada *packet loss* Wi-Fi karena kapasitas *bandwidth* seluler lebih kecil daripada kapasitas *bandwidth* Wi-Fi.

BAB 5

PENUTUP

5.1 Kesimpulan

Setelah dilakukan perancangan dan pengujian sistem pemantau posisi armada transportasi umum (bus) Surabaya, maka secara keseluruhan dapat diambil kesimpulan sebagai berikut:

1. Posisi armada dapat diketahui secara realtime dengan mengintegrasikan database dari server dengan GPS yang tertanam pada setiap armada
2. Dengan membandingkan acuan dan pengukuran dapat dilihat rata-rata jarak error yang dapat terjadi sekitar 14 meter.
3. GPS Sipil yang dipergunakan masyarakat sipil mempunyai tingkat error yang cukup tinggi
4. Pengujian *packet loss* terkecil pengiriman data melalui jaringan wi-fi terletak pada ukuran data 250, 450, dan 550 *byte* sebesar 0.26667% (*perfect*). Sedangkan pada jaringan seluler nilai *packet loss* terkecil terletak pada ukuran data 450 *byte* sebesar 0.13333% (*perfect*)
5. Kemampuan monitoring yang dapat dihasilkan pada penelitian ini adalah berjalannya 5 armada bus disaat yang bersamaan dengan tujuan yang berbeda.

5.2 Saran

Adapun hal-hal yang masih bisa dikembangkan dari rancang bangun pemantau posisi armada ini adalah:

1. Pengembangan aplikasi berbasis web yang telah ada sehingga mampu menampilkan lebih detail informasi pada armada seperti sisa bahan bakar, jumlah penumpang yang ada serta waktu naik dan turunnya penumpang.
2. Kedepannya dapat dilakukan pengukuran terhadap *troughput* dan *jitter* pada sistem yang telah ada.
3. Penggunaan GPS militer agar lebih presisi dalam menentukan posisi

Halaman ini sengaja dikosongkan

Daftar Pustaka

- [1] Affandi Achmad dan tim. “Laporan Akhir Kajian Sistem dan Teknologi IT dalam Rangka Integrasi Angkutan Massal Cepat *Trunk* dan *Feeder* AMC Surabaya,” Dinas Perhubungan Kota Surabaya, Surabaya, 2015.
- [2] Yulianto, Prasetyo. “Rancang Bangun Protokol E-Ticketing,” Institut Teknologi Sepuluh Nopember, Surabaya, 2015
- [3] Setijadi, Eko dan tim. “Laporan Akhir Detail Engineering Design Manajemen Armada Angkutan Massal Cepat Surabaya,” Dinas Perhubungan Kota Surabaya, Surabaya, 2015.
- [4] Indrajaya, Muhammad Aristo. “Perancangan *Geographic Information System* untuk *Tracking* dan *Routing* Menggunakan Algoritma Dijkstra pada Kendaraan Angkutan Umum,” Institut Teknologi Sepuluh Nopember, Surabaya, 2015
- [5] Halim, Ansarullah. Affandi, Achmad. Maulidiyanto, Achmad. “Analisa Trafik Komunikasi Data pada Kanal Propagasi VHF Bergerak untuk Kelautan,” Institut Teknologi Sepuluh Nopember, Surabaya, 2011.
- [6] Deifitte, Prisselort. “Perancangan alat ukur ketinggian menggunakan GPS, mikrokontroler dan visual basic,” Universitas Indonesia, Depok, Indonesia, 2010.
- [7] ITU-T Series G, “*Transmission system and media, digital system and network*,” 2003.
- [8] Zhang, Yongqiang. Li, Liangliang. Zhang, Yongjian “Design and Implementation of GPS/DR/GPRS Integrated Position and Monitoring System for Vehicle,” University of Engineering, Handan, China, 2009.
- [9] S A, Mulay. C S, Dhekne. R M, Bapat. T U, Budukh. S D, Gadgil “Intelligent City Traffic Management and Public Transportation System”, Departement of Computer, PVG’s COET Pune, India, 2011.

Halaman ini sengaja dikosongkan

Lampiran

a. Proposal Tugas Akhir

Departemen Teknik Elektro
Fakultas Teknologi Elektro - FTI

TE141599 TUGAS AKHIR – 4 SKS

13 FEB 2017

Nama Mahasiswa : Muhammad Hilmy Albanna
Nomor Pokok : 2213 100 050
Bidang Studi : Telekomunikasi Multimedia
Tugas Diberikan : Semester Genap 2016/2017
Dosen Pembimbing : 1. Dr. Ir. Achmad Affandi, DEA
2. Dr. Ir. Endroyono, DEA

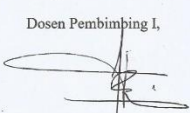
Judul Tugas Akhir : Rancang Bangun Pemantau Posisi Armada Transportasi Umum pada Smart Transportation
(Design of Public Transportation Fleet Monitoring Position on Smart Transportation)

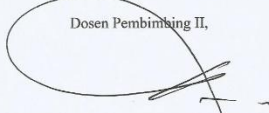
Uraian Tugas Akhir :


Teknologi informasi merupakan salah satu solusi untuk pengelolaan transportasi di Kota Surabaya untuk meningkatkan keefektifan pelayanan publik. Salah satunya adalah dengan cara penggunaan On-Board Unit (OBU) pada Angkutan Massal Cepat yang akan diterapkan di Kota Surabaya. OBU adalah terminal elektronik yang dapat memberikan informasi mengenai identitas dan posisi AMC, dan menghubungkan komunikasi antara CC-ROOM dengan Pengemudi. Tugas akhir ini bertujuan untuk merancang sebuah sistem yang dapat menerima data dari GPS yang berada di OBU untuk pemantauan dari semua armada.


Perancangan dan implementasi dilakukan dalam 2 tahap, yaitu membuat aplikasi sistem monitoring antara armada dan terminal serta memanfaatkan google API dalam menunjang penentuan titik posisi armada yang akan dikirimkan ke CC-ROOM setiap beberapa detik. Aplikasi pemantauan posisi tiap armada dibuat menggunakan Borland Delphi 7. Jaringan yang digunakan yaitu jaringan seluler.

Pengujian sistem pemantauan posisi armada dilakukan dengan dua macam pengujian, yaitu pengujian fungsi dari sistem yang telah dibuat kemudian pengujian dari sisi performansi yang dapat dilihat dari kelancaran penerimaan data GPS untuk menentukan posisi dari armada transportasi massal cepat tersebut.

Dosen Pembimbing I,

Dr. Ir. Achmad Affandi, DEA
NIP. 196510141990021001

Dosen Pembimbing II,

Dr. Ir. Endroyono, DEA
NIP. 196504041991021001

Mengesahului,
Ketia Program Studi SI

Dedet C. Riwanto, ST, M.Eng., Ph.D.
NIP. 197311192000031001

Menyetujui,
Kepala Laboratorium Jaringan Telekomunikasi

Dr. Ir. Achmad Affandi, DEA
NIP. 196510141990021001

b. Listing Program

Loggin:

```

<section>
  <div class="container-alt">
    <div class="row">
      <div class="col-sm-12">
        <div class="wrapper-page">
          <div class="m-t-40 p-20 account-pages">
            <div class="text-center">
              <h1 class="text-uppercase">
                SITS
              </h1>
              <!--<h4 class="text-uppercase font-bold m-b-0">Sign
In</h4>-->
            </div>
            <div class="account-content">
              <form method="post" class="form-horizontal" >
                <div class="form-group ">
                  <div class="col-xs-12">
                    <input class="form-control" name="user"
type="text" required="" placeholder="Username">
                  </div>
                </div>
                <div class="form-group">
                  <div class="col-xs-12">
                    <input class="form-control" name="password"
type="password" required="" placeholder="Password">

```

```

10">
        <div class="col-xs-12">
            <button class="btn w-md btn-bordered btn-danger
waves-effect waves-light" type="submit">Log In</button>

```

Monitoring armada

```

<?php
$host = "localhost";

$db_name = "ta01lawawa_gps";

$username = "ta01lawawa_gps";

$password = "bHON@~R6fG08";

//$db_name = "gps";

//$username = "root";

//$password = "";

$conn = null;

try {

    $conn = new PDO("mysql:host=" . $host . ";dbname=" . $db_name,
$username, $password);

    $conn->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);

} catch (PDOException $exception) {

    echo "Connection error: " . $exception->getMessage();

}

<script type="text/javascript">

    $('#bus-img').hide();

    var map;

```

```

var i = 0;
var markers = [];
var t;
var active = 1;
var id = 0;
function flags() {
    document.location='<?php echo URLS ?>/flags/'+id+'/1';
function initMap() {
    $('#bus-img').hide();
    var myLatLng = {lat: -7.350730, lng: 112.76632};
    var map = new google.maps.Map(document.getElementById('gmaps-
route'), {
        zoom: 12,
        center: myLatLng
<?php
    try {
        $sql = "SELECT * FROM lokasi WHERE flag = 0";
        $stmt = $conn->prepare($sql);
        $stmt->execute();
        while ($array = $stmt->fetch(PDO::FETCH_ASSOC)) {
            var loc<?php echo $array['id'] ?> = {lat: <?php echo $array['lat'] ?>,
lng: <?php echo $array['lang'] ?>};
            var marker<?php echo $array['id'] ?> = new google.maps.Marker({
                position: loc<?php echo $array['id'] ?>,
                map: map,

```

```

        icon: 'bus2.png',
        id: <?php echo $array['id'] ?>
    <?php
    } catch (PDOException $exception) {
    echo "Select error: " . $exception->getMessage();
    (function worker() {
        if (i == 500) {
            location.reload();

            $.ajax({
                cache: false,
                url: '<?php echo URLS ?>/getlatlang',
                success: function (data) {

                    <?php
                    try {

                        $sql1 = "SELECT * FROM lokasi WHERE flag = 0";
                        $stmt1 = $conn->prepare($sql1);
                        $stmt1->execute();

                        while ($array = $stmt1->fetch(PDO::FETCH_ASSOC))
                        loc<?php echo $array['id'] ?> = {

                            lat: parseFloat(data[<?php echo $array['id'] - 1 ?>].lat),
                            lng: parseFloat(data[<?php echo $array['id'] - 1 ?>].lang)

                            if (i == 0) {

                                marker<?php echo $array['id'] ?> = new
                                google.maps.Marker({

                                    position: loc<?php echo $array['id'] ?>,

```

```

        map: map,
        icon: 'bus2.png',
        id: <?php echo $array['id'] ?>
    } else {
        movemarker<?php echo $array['id'] ?>(loc<?php echo
$array['id'] ?>);
        <?php
    } catch (PDOException $exception) {
        echo "Select error: " . $exception->getMessage();
    complete: function () {
        // Schedule the next request when the current one's complete
        t = setTimeout(worker, 2000);
    <?php
    try {
        $sql2 = "SELECT * FROM lokasi WHERE flag = 0";
        $stmt2 = $conn->prepare($sql2);
        $stmt2->execute();
        while ($array = $stmt2->fetch(PDO::FETCH_ASSOC)) {
            ?>
            google.maps.event.addListener(marker<?php echo $array['id'] ?>,
'click', function () {
                $.ajax({
                    cache: false,
                    url: '<?php echo URLS ?>/getbus/' + marker<?php echo
$array['id'] ?>.id,
                    complete: function (data) {

```

```

id = <?php echo $array['id'] ?>;
$('#bus-img').show();

// Schedule the next request when the current one's complete
$('#tujuan').html(data.responseJSON.tujuan);
$('#berangkat').html(data.responseJSON.berangkat);
$('#plat').html(data.responseJSON.plat_nomor);
$('#jadwal').html(data.responseJSON.jadwal);
$('#bus-img').attr('src', data.responseJSON.gambar);

<?php
} catch (PDOException $exception) {
echo "Select error: " . $exception->getMessage();
<?php

try {
$sql3 = "SELECT * FROM lokasi WHERE flag = 0";
$stmt3 = $conn->prepare($sql3);
$stmt3->execute();

while ($array = $stmt3->fetch(PDO::FETCH_ASSOC)) {
?>

function movemarker<?php echo $array['id'] ?>(location) {
    marker<?php echo $array['id'] ?>.setPosition(new
google.maps.LatLng(location));
<?php
}

} catch (PDOException $exception) {

```



```

    echo "Select error: " . $exception->getMessage();

    // Sets the map on all markers in the array.

    function setMapOnAll(map) {

        for (var i = 0; i < markers.length; i++) {

            markers[i].setMap(map);

            // Removes the markers from the map, but keeps them in the array.

            function clearMarkers() {

                setMapOnAll(null);

            }

        }

    }

</script>

<!--<script async defer-->

<!--
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyDcGaIUeXmbV
7eJ686ng3n7XbIqvef77Vk&callback=initMap">-->

<!--//<!--
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyDcGaIUeXmbV
7eJ686ng3n7XbIqvef77Vk&callback=initMap">-->

<!------>

<!--</script>-->

<script async defer
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyDcGaIUeXmbV
7eJ686ng3n7XbIqvef77Vk&callback=initMap">

</script>

```

Update Lokasi

```

$app->post('/postlatlang', function () use ($app, $conn) {

    try {

        $id = $_POST['id'];
    }

```

```

$lat = $_POST['lat'];
$lang = $_POST['lang'];
$data = [
    "lat" => $lat,
    "lang" => $lang,
];
$s = "";
$i = 0;
foreach ($data as $key => $values) {
    $s .= $key . "=" . $values . "" . ($i < (count($data) - 1) ? "," : "");
    $i++;
}
$sql = "UPDATE lokasi set $s where id = $id ";
$stmt = $conn->prepare($sql);
$stmt->execute();
$t = microtime(true);
$micro = sprintf("%06d", ($t - floor($t)) * 1000000);
$d = new DateTime( date('Y-m-d H:i:s.'.$micro, $t) );
print $d->format("Y-m-d H:i:s.u");

$data = [
    "lat" => $lat,
    "lang" => $lang,
    "bus" => $id,
    "waktu" => $d->format("Y-m-d H:i:s.u"),

```

```

"terima" => $_POST['terima'],
"size" => $_SERVER['CONTENT_LENGTH'],
];
$columns = implode(" ", array_keys($data));
$s = "";
$i = 0;
foreach ($data as $key => $values) {
    $s .= "" . $values . "" . ($i < (count($data) - 1) ? ", " : "");
    $i++;
}
$sql = "INSERT INTO log_kendaraan ($columns) VALUES ($s)";
$stmt = $conn->prepare($sql);
$stmt->execute();
} catch (PDOException $exception) {
    echo "Connection error: " . $exception->getMessage();
}
echo $sql;

```

Log Perpindahan Bus

```

<div class="row">
    <div class="col-sm-12">
        <div class="card-box table-responsive">
            <table id="datatable-responsive"
                class="table table-striped table-bordered dt-responsive
nowrap" cellpadding="0"
                width="100%">

```

```

        <th class="col-md-1">Waktu Terima</th>
        <th class="col-md-1">Waktu Kirim</th>
        <th class="col-md-1">Delay</th>
        <th class="col-md-1">Size</th>
        <th class="col-md-3">Lat,Lang</th>
    </?php
    $host = "localhost";
    //      $db_name = "gps";
    //      $username = "root";
    //      $password = "";
    $db_name = "ta01lawas_gps";
    $username = "ta01lawas_gps";
    $password = "bHON@~R6fG08";
    $conn = null;
    try {
        $conn = new PDO("mysql:host=" . $host .
";dbname=" . $db_name, $username, $password);
        $conn->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);
    } catch (PDOException $exception) {
        echo "Connection error: " . $exception-
>getMessage();
    }
    </?php
    try {
        $i = 0;

```

```

= $_GET[id] ";

$sql = "SELECT * FROM log_kendaraan where bus

$stmt = $conn->prepare($sql);

$stmt->execute();

while ($array = $stmt-
>fetch(PDO::FETCH_ASSOC)) {

    $i++;

    <td> <?php echo $array['terima'] ?></td>

    <td> <?php echo $array['waktu'] ?></td>

    <?php

        $r1= explode(".", $array['terima']);

        $r2 = explode(".", $array['waktu']);

        $detik1=explode(":", $r1[0]);

        $detik2=explode(":", $r2[0]);

        $jam1=explode(" ", $detik1[0]);

        $jam2=explode(" ", $detik2[0]);

        <td> <?php echo
round(((( $jam1[1]."".$detik1[1]."".$detik1[2]."".$r1[1])-
($jam2[1]."".$detik2[1]."".$detik2[2]."".$r2[1]))/1000) ?></td>

        <td> <?php echo $array['size']

        <td><?php echo $array['lat'] ?>,<?php echo
$array['lang'] ?></td>

    } catch (PDOException $exception) {

        echo "Select error: " . $exception->getMessage();

```

Android Studio

```

public class LocationService extends Service {
    private LocationManager locationManager;
    private MyLocationListener myLocationListener;
    public static double latitude = 0;
    public static double longitude = 0;
    public static TextView trwy;

    @Override
    public IBinder onBind(Intent intent) {
        return null;
    }

    @Override
    public void onCreate() {
        super.onCreate();
        this.locationManager = (LocationManager)
this.getSystemService(Context.LOCATION_SERVICE);
        this.myLocationListener = new MyLocationListener();
        if (ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED &&
ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
            return;
        }

        locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER,
500, 10, myLocationListener);

    }

    @Override
    public void onDestroy() {
        super.onDestroy();
        try {

            if (ActivityCompat.checkSelfPermission(this,

```

```

Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED &&
ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {

```

```

    return;
}
locationManager.removeUpdates(myLocationListener);
} catch (Exception e) {
    e.printStackTrace();
}
}

```

```

public class MyLocationListener implements LocationListener {
    @Override
    public void onLocationChanged(Location location) {

```

```

RequestHandler rh = new RequestHandler();
    latitude = location.getLatitude();
    longitude = location.getLongitude();
    trwy.setText(latitude + "," + longitude);
    EventBus.getDefault().post(location);

    Log.d("lam", "onLocationChanged:lam " + location.getLatitude());
    Log.d("lam", "onLocationChanged:lam " + location.getLongitude());
    sendImage();
}

```

```

@Override
    public void onStatusChanged(String provider, int status, Bundle
extras) {
    }
    @Override
    public void onProviderEnabled(String provider) {
    }
    @Override
    public void onProviderDisabled(String provider) {

```

```

    } b
}
private void sendImage() {
    class UploadImage extends AsyncTask<Bitmap, Void, String> {

        RequestHandler rh = new RequestHandler();

        @Override
        protected void onPreExecute() {

            @Override
            protected void onPostExecute(String s) {
                super.onPostExecute(s);
            }

            @Override
            protected String doInBackground(Bitmap... params) {

                HashMap<String, String> data = new HashMap<>();
                data.put("id", String.valueOf(1));
                data.put("lat", String.valueOf(latitude));
                data.put("lang", String.valueOf(longitude));
                String result =
rh.sendPostRequest("http://a269a8e7.ngrok.io/gps/postlatlang", data);

                Log.d("lam", "onLocationChanged:lam " + result);
                return result;
            }
        }
    }

    UploadImage ui = new UploadImage();
    ui.execute();
}

```


RIWAYAT PENULIS



Muhammad Hilmy Albanna, lahir di Sidoarjo pada tanggal 4 Desember 1994. Menamatkan pendidikan dasarnya di SD Muhmmadiyah 1 Sidoarjo pada tahun 2007 lalu meneruskan di SMP Negeri 5 Sidoarjo dan lulus pada tahun 2011. Kemudian melanjutkan pendidikan menengah atas di SMA Negeri 1 Sidoarjo dan lulus pada tahun 2013. Setelah itu melanjutkan pendidikan strata satu (S1) di Departemen Teknik Elektro Institut Teknologi Sepuluh Nopember Surabaya pada tahun 2013 hingga 2017 dengan program studi

Telekomunikasi Multimedia. Penulis dapat dihubungi melalui alamat email : hilmyalbanna2@gmail.com.